

2–'88

DAS COMPUTERMAGAZIN FÜR AMIGA-FANS

Test: Turbo-Karte

Der schnellste Amiga

Die besten Utilities

Kopierprogramme im Vergleich

Test: C 64-Emulatoren

Der C 64 im Amiga

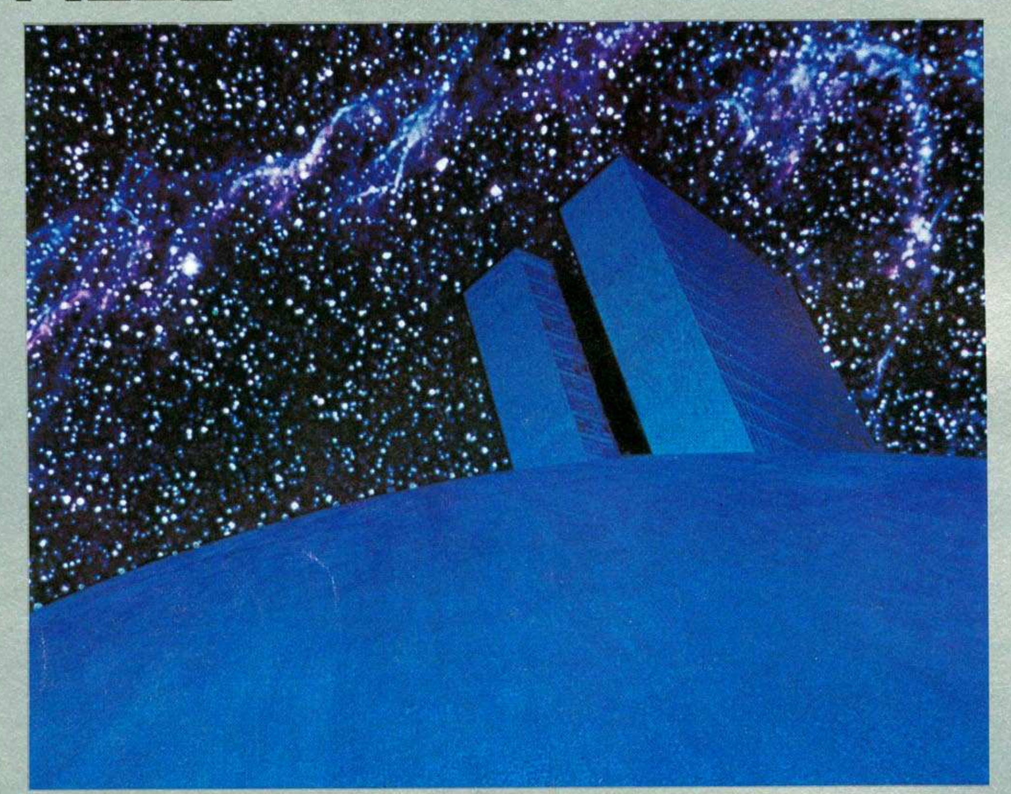
Die besten Musikprogramme

Amiga in Concert

TURBO

Das AMIGA-Magazin testet: Neue Software: Maxiplan, Silver, Butcher 2.0

AN ALLE AMIGA-BESITZER:



Entdecken Sie neue Welten auf Ihrem Amiga!

The 64 Emulator erweitert den Horizont Ihres Amiga um die abertausende von Programmen, die für den Commodore 64 geschrieben wurden. Ja, die erprobten Textverarbeitungen, Datenbänke, Rechenblätter und aufregenden Spiele können ab sofort auf Ihrem Amiga laufen.

Um alle Vorteile der Hardware Ihres Amiga zu nutzen, wurde dieser Emulator 100%ig in 68000 Maschinensprache geschrieben. Hierdurch erhalten Sie maximale Geschwindigkeit und die Unterstützung aller Vorteile des C 64 inklusive Farbe, Grafik und Sound. Durch einen Modus können Sie die Farbe abschalten und einen Geschwindigkeitszuwachs erlangen.

Alle Amiga-Laufwerke und Drucker werden voll unterstützt. Durch die Transfersoftware ist es kinderleicht, C-64-Programme auf Amiga-Disketten zu übertragen. Für gesteigerte Kompatibilität mit geschützter Software verbindet das serielle Interface jedes C-64 Laufwerk und Drucker mit Ihrem Amiga.

Warum auf ein erweitertes Amiga-Software Angebot warten, wenn Sie sofort Qualitäts-Software haben können, die schon von 6.000.000 Computer-Besitzern benutzt wird. **The 64 Emulator** von Readysoft ...
hier, jetzt, für Sie.
Händleranfragen erwünscht.
Telefonische Bestellung ab sofort möglich.

THE 64 EMULATOR FÜR IHREN AMIGA DM 148,-

JA, ich will die tausende von C-64 Programmen auf meinem Amiga benutzen.

☐ **THE 64 Emulator** mit seriellem Interface DM 148,—

Für welchen Computer? ☐ **Amiga 500** ☐ **Amiga 1000** ☐ **Amiga 2000**

Zahlungsweise: ☐ **Scheck** ☐ **Nachnahme** ☐ **Bar**

Bitte legen Sie DM 8,— für Porto und Verpackung bei.

Name ______________________

Straße, Nr. ______________________

PLZ, Wohnort ______________________

Bitte senden an:

CompuStore
Fritz Reuter Str. 6
6000 Frankfurt/Main

oder:

Atlantis
Dunantstraße 53
5030 Hürth

ATLANTIS

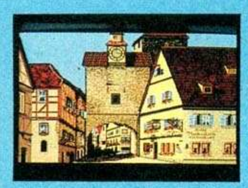
CompuStore
Handelsgesellschaft mbH für Hard- und Software
Fritz-Reuter-Straße 6
6000 Frankfurt/Main 1 (West Germany)
Telefon (069) 56 73 99

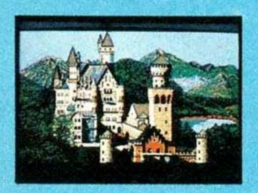
Unsere

SPIELE DES JAHRES

für den

COMPUTER DES JAHRES





EMETIC SKIMMER

von Joachim Deeg

Wir schreiben das Jahr 5354. Die Erde hat soeben den ersten Computer mit künstlicher Intelligenz fertiggestellt und hat ihn auf dem fernen Planeten Pluto stationiert. Aber die Wissenschaftler haben die Fähigkeiten des Computers unterschätzt. Mit seinem immensen Wissen und seinem fehlenden Gewissen stellt er eine ernstzunehmende Gefahr für das gesamte Sonnensystem dar. Ihre Aufgabe ist es nun, den Emetic Skimmer, ein Raumschiff mit Spezialausstattung, durch die Verteidigungslinien des Computers zu steuern und ihn am Ende zu zerstören. Das neueste Superspiel für den Amiga - 2 Disketten randvoll gepackt mit Spitzen-Grafik, Digisounds und jeder Menge Action.

49.95





© SE-DESIGN 0241-151034

MIKE - THE MAGIC DRAGON

von Frank Neuhaus

Mike ist ein kleiner Drachenjunge, der von dem bösen Professor Dragan Drachenklau entführt worden ist, denn dieser benutzt in seinen Burgen und Schlössern mit Vorliebe kleine Drachen für seine garstigen Experimente. Doch Mike hat sich von den Ketten befreien können und sucht nun einen Weg durch das Labor des Professors, der in die Freiheit führt. Ein Kletter- und Springspiel der Extraklasse mit vielen neuartigen Extras.











PHALANX II - THE RETURN

nach Oliver Czesla

Nach dem Riesenerfolg des Actiongames PHALANX ist der langersehnte Nachfolger jetzt fertig. Und das Warten hat sich gelohnt, denn Grafik, Sound und Action haben eine weitere Steigerung erfahren. 1 oder 2 Spieler (gleichzeitig!) rasen über die vertikal scrollende Landschaft auf der Suche nach immer neuen Formationen der Gegner.



SIE GLAUBEN KEINEN WERBESPRÜCHEN?!? DANN LESEN SIE SELBST, WAS DIE DEUTSCHE FACHPRESSE ÜBER KINGSOFT-SPIELE SCHREIBT:

CHALLENGER von Andreas von Lepel

„Challenger ist kein Spiel, das man zur Spitzenklasse zählen kann, trotzdem ist es wegen seines relativ niedrigen Preises zu empfehlen." (ASM) **29.95 DM**

CITY DEFENCE von Uwe Kühner

„Originalgetreue Grafik, guter Sound, fantastische Steuerung und, was vor allem wichtig ist, eine gute Spielatmosphäre. Da es für den Amiga im Moment außerdem an guten Action-Spielen mangelt, kann man City Defence also nur wärmstens empfehlen, denn es ist wirklich Klasse!" (ASM) **29.95 DM**

CRUNCHER FACTORY von M. Aberle

„Vor allem zu zweit macht Cruncher Factory Spaß und angesichts des sehr attraktiven Preises kann man das Spiel empfehlen." (HAPPY COMPUTER)

„Eine amüsante und schnelle Pacman-Variante ist Cruncher, der zudem noch mit einem Level-Editor versehen ist. ... Die Grafik ist ausreichend gut, die digitalisierte Titelmusik wieder einmal hörenswert. Wer diesen Klassiker gerne für den Amiga haben möchte, sollte die Gelegenheit wahrnehmen." (KICKSTART) **29.95 DM**

DEMOLITION von Martin Aberle

„Von der Grafik her ist das Spiel recht gut, auch der digitalisierte Sound ist hörenswert." (KICKSTART) **29.95 DM**

EMERALD MINE

von Klaus Heinz & Volker Wertich

„Emerald Mine ist zur Zeit eines der besten (wenn nicht sogar das beste!) für den Amiga erhältliche Spiel. Da soll noch mal einer behaupten, daß in Deutschland keine guten Spiele produziert werden. Emerald Mine hebt sich von den bisher vorherrschenden Shoot 'em up Games deutlich ab und setzt in der Amiga-Softwareszene neue Maßstäbe. Dafür sorgt auch der unglaubliche Preis von nur knapp 30.- DM." (AMIGA SPECIAL)

„Die Steuerung des (oder der) Männchen(s) erfolgt mit dem Joystick und ist sehr exakt. So richtig Spaß macht das Game dann, wenn man zu zweit spielt. Zum Programm Emerald Mine kann man Kingsoft nur gratulieren, was einem bei diesem Spiel alles geboten bekommt, das sucht seinesgleichen. Nimmt man noch den überaus günstigen Preis von ca. 30 DM, dann ist mit Emerald Mine ein richtiger Hit entstanden, der jede Spielesammlung bereichert." (ASM)

„Die flotte Grafik mit witzigen Monster-Sprites und vor allem der Sound gefallen mit sehr gut: Die fetzige Titelmusik und die digitalisierten Sound-Effekte sind exzellent. Das Verblüffendste an diesem tollen Spiel ist aber der sehr niedrige Preis." (HAPPY COMPUTER)

„Grafik und Scrolling dieses Spieles sind überzeugend. ... Insgesamt hinterließ Emerald Mine einen guten Eindruck." (KICKSTART) **29.95 DM**

FLIP FLOP Amiga-Version von Edgar Törnig

„Das Programm ist für meinen Geschmack fast schon zu gut. Ich will ja nicht von mir behaupten, daß ich gut Reversi spielen kann, aber selten zuvor bin ich mit solch fliegenden Fahnen untergegangen." (ASM) **29.95 DM**

FORTRESS UNDERGROUND

von Andreas von Lepel

„Die Steuerung des Hubschraubers ist sehr exakt, in dieser Hinsicht ist das Programm gelungen. Auch das Scrolling geht in Ordnung. Der Sound kann sich hören lassen." (ASM) **29.95 DM**

KARATE KING von Marc Ebner

„Im Vergleich zu den beiden anderen Karatespielen ist Karate King das schnellste." (KICKSTART) **49.95 DM**

PINBALL WIZARD

von Hans Georg Berg

„Computer-Flippern stehe ich eigentlich skeptisch gegenüber, weil der Spielablauf größtenteils vom Zufall abhängt. Pinball Wizard ist aber so clever gemacht, daß er mich zu längeren Flipper-Sessions verführen kann. Die schön gezeichnete Grafik und die Digi-Sounds können überzeugen." (HAPPY COMPUTER)

„Die Idee, auf einem Homecomputer einen Flipper zu programmieren, ist zwar nicht neu, aber ich glaube, daß dem Programmierer von Pinball Wizard ein großer Wurf gelungen ist. ... Pinball Wizard ist die realistischste Umsetzung dieses Spiels, die ich kenne. ... Die Grafik, die dem Spieler geboten wird, ist vom Feinsten. ... Auch nach längerer Spieldauer konnte ich keine Schwachpunkte entdecken, so daß ich dieses Programm unbedingt empfehlen kann. Für mich ist Pinball Wizard ein Spiel, das in keiner (Amiga-) Sammlung fehlen darf." (ASM) **49.95 DM**

QUIWI Amiga-Version von Andreas Arens

„Fazit: ein sehr gutes Computer-Gesellschaftsspiel mit Zukunft." (HAPPY COMPUTER)

„Sogar die zehnte Revancherunde macht noch Spaß, denn 4000 Fragen machen Wiederholungen selten. QUIWI ist ein amüsantes Quizspiel für Feste und Familienfeiern." (RUN)

„QUIWI hingegen ist ein wirklich spaßiges wie unterhaltsames und lehrreiches Programm, wenn man mir diese Wiederholung von vorher verzeihen mag. Empfehlenswert!" (ASM)

„Ein abwechslungsreiches Spiel für die ganze Familie." (CHIP) **49.95 DM**

SOCCER KING

von Wolfgang & Helmut Theis

Fußballspiel für 1/2 Spieler; noch nicht getestet. **49.95 DM**

SPACE BATTLE von Andreas Hommel

„Space Battle ist ein solide gemachtes Spiel, das trotz der alten Spielidee viel Spaß bringt. Wenn man bedenkt, daß sowohl Cruncher Factory als auch Space Battle unter 30 Mark kosten, dann fällt einem die Entscheidung zum Kauf nicht mehr schwer. ... Beide haben, obwohl sie billig sind, mehr zu bieten als ähnliche Spiele." (AMIGA-MAGAZIN)

„Für knapp dreißig Mark erhält man mit Space Battle zwar kein neuartiges Spiel. Trotzdem ist es, besonders im Competition-Modus, spannend." (KICKSTART) **29.95 DM**

TYPHOON

von Christoph Sing & Rolf Wagner

„Auf diesem Hintergrund tummeln sich Massen von Sprites, die hervorragend ausgearbeitet und animiert sind. Es macht wirklich Laune, die verschiedenen Varianten von Raumschiffen und Hindernissen zu betrachten. ... Trotz einiger taktischer Varianten ist TYPHOON ein reinrassiges Actionspiel, das die Fans dieses Genres begeistern dürfte. Besonders die 50 verschiedenen Screens mit der erstklassigen Grafik und den blitzsauber animierten Sprites garantiert Freunden des Actionspiels lange Spielmotivation. ... Tolle Grafik und viel Action in TYPHOON ergänzen sich zu einem unterhaltsamen Spiel." (68000er)

„Wenn mich zukünftig jemand fragt, was ich unter einem guten Action-Spiel verstehe, werde ich ihn an TYPHOON verweisen. Die Handlung ist zwar dürftig bis nicht vorhanden, aber die Grafik ist ungeheuer faszinierend. ... Die Asteroiden-Felder, der Dschungel und die Stadt haben mich sofort begeistert. Allein die Neugierde, welche Sprites in der nächsten Angriffswelle kommen, hat mich vor den Monitor gebannt. ... Die Grafik steht so manchem Spielautomaten in nichts nach. ... Das Scrolling ist samtweich, Sprites sind reichlich vorhanden und gut animiert. Der Schwierigkeitsgrad des Spiels steigert sich ganz allmählich und die Idee mit den drei Extra-Waffen weist die Monotonie in die Schranken, die sonst nach einer Weile bei Ballerspielen entsteht. Die Präsentation kann sich auch sehen lassen: Schickes Titelbild, digitalisierte Disco-Rhythmen und eine prächtige High-Score-Liste." (HAPPY COMPUTER) **49.95 DM**

WILLY THE KID von Peter Meyer

Deutsches Text-/Grafik-Adventure; noch nicht getestet. **29.95 DM**



Alle Preise sind unverbindliche Preisempfehlungen. Bei uns gibt's jeden Monat Neuerscheinungen für Ihren AMIGA - am besten unseren Katalog anfordern!

SPITZEN-SOFTWARE
MADE IN GERMANY



Grüner Weg 29 · D-5100 AACHEN
☎ 0241/15 20 51 · Fax 0241/15 20 54

KINGSOFT-Produkte erhalten Sie unter anderem in ausgewählten Fachabteilungen von:











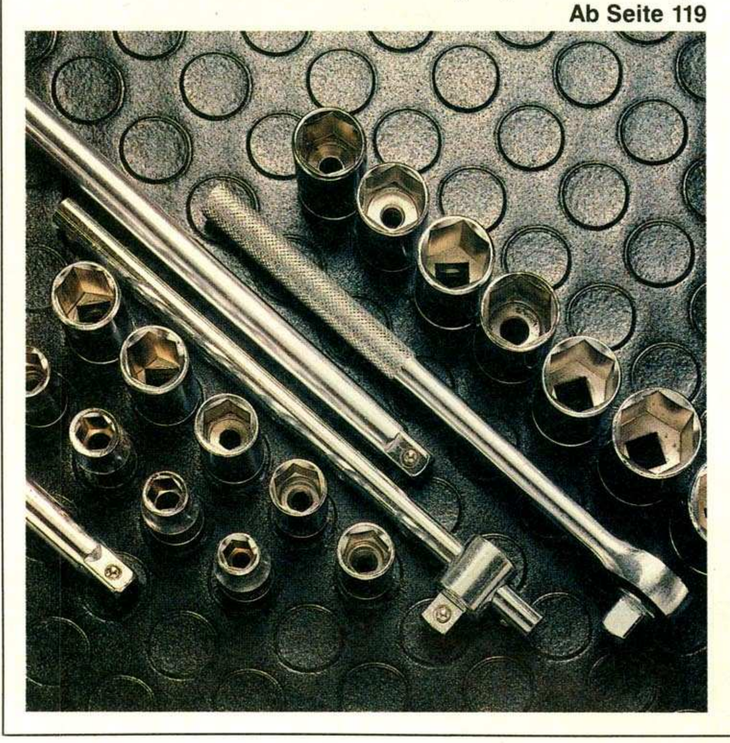




Das Beste aus der Werkzeugkiste

Hilfsprogramme, auch als Tools und Utilities bezeichnet, sind das Salz in der Suppe des Computer-Anwenders. Wir haben die besten Tools für Sie herausgepickt und helfen Ihnen bei der Wahl der geeigneten Werkzeuge.

Ab Seite 119



Turbo-Amiga startet durch

Mit der Turbo-Karte können Sie den Amiga beschleunigen. Beim Raumfahrt-Unternehmen MBB-ERNO wird der schnelle Amiga bereits eingesetzt. Lesen Sie, zu welchen Leistungen er fähig ist.

Seite 28

Foto: MBB-ERNO



AKTUELL

EINSTEIGER

MUSIK

SOFTWARETEST

HARDWARETEST

KURSE

TIPS & TRICKS

SOFTWARE

INHALT 2/88

Mit Musik geht alles besser

Interessieren Sie sich für Noten und Instrumente? Wollten Sie schon immer einmal wissen, wie Sampling überhaupt funktioniert? Dann ist der Musikschwerpunkt mit Grundwissen und Tests das Richtige für Sie.
Ab Seite 21



HARDWARE

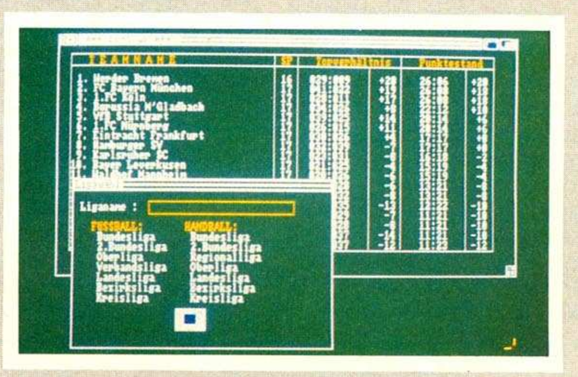
HILFSPROGRAMME

AUFRUFE UND WETTBEWERBE

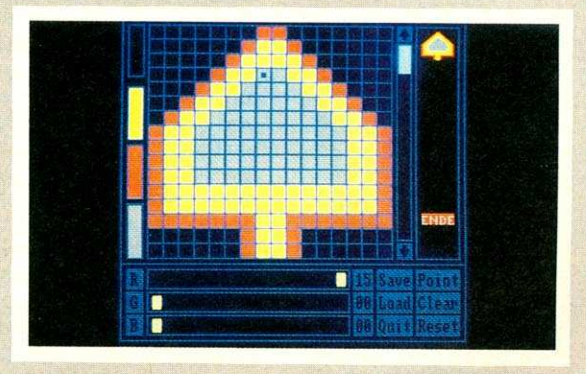
RUBRIKEN

Dieses Symbol zeigt an, welche Programme auf Diskette erhältlich sind.

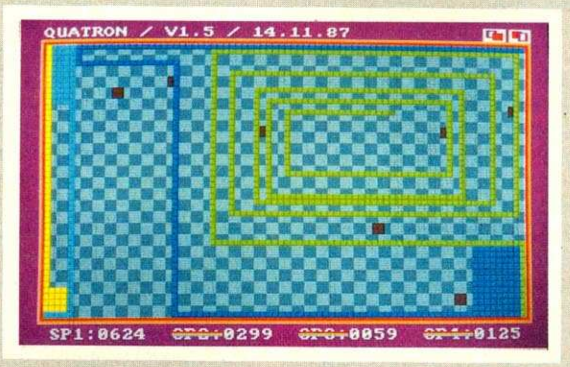
LISTINGS ZUM ABTIPPEN



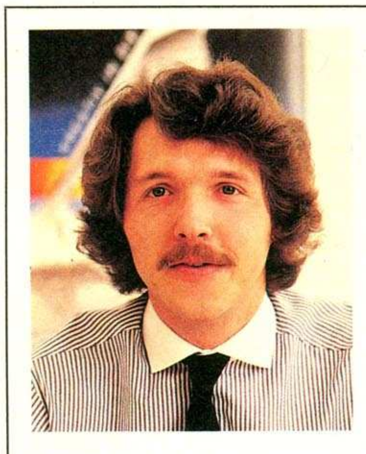
»Quatron«, ein Spiel mit Supergrafik und großer Spielmotivation, für bis zu 3 Spieler



Eigene Mauszeiger in Basic-Programmen sind kein Problem mit »MouseCreator«



Das Programm des Monats: Mit »LigaTab« verschiedene Tabellen verwalten



Laßt Zahlen sprechen

Commodore hat allein in den Monaten Oktober bis Dezember '87 fünfzigtausend Amiga 500 verkauft. Das sind mehr Einheiten als im ersten Jahr seit der Markteinführung. Es hätte noch besser laufen können, wenn das Unternehmen ob der überraschenden Nachfrage nicht von Lieferschwierigkeiten heimgesucht worden wäre. Insgesamt haben Ende 1987 rund 130000 Amiga 500, 1000 und 2000 in Deutschland ihre stolzen Besitzer gefunden. Das ist doch eine deutliche Absage der kreativen Amiga-Fans an den MS-DOS-Standard — den seine Verfechter immer noch so gerne als das Nonplusultra in der Computerei ansehen.

Man hört den einen oder anderen dieser Sicherheitsapsteln fragen: »Viele Farben hat er ja, aber wie sieht es denn mit der Software aus?« Gut sieht es mit der Software aus. Der neueste Softwarekatalog von Commodore beweist es. Mehr als tausend Programme und Hardware-Erweiterungen sind dort aufgeführt. Wenn da nicht jeder die richtige Software findet.

Was macht Atari? In der Brust der STs schlägt das gleiche Herz wie im Amiga. Von daher ist eine gewisse Verwandtschaft nicht zu leugnen. Doch die Vertreter der alternativen Computer sind sich spinnefeind. Atari und Commodore konnten sich nie so richtig ausstehen, obwohl beide mit kreativen Ideen und deren Umsetzung eigentlich gegen den von IBM gesetzten Standard ständig angingen. Statt zu versuchen, zusammen mit Apple, die im Macintosh auch einen 68000-Prozessor arbeiten lassen, eine alternative Front gegen »verstaubte« Großkunden-Computer zu eröffnen, verzettelt man sich in energieverschleißende Grabenkämpfe.

Als ginge es darum, einen potentiellen Computer-Freak davon überzeugen zu müssen, daß nicht die richtige Seite auf dem Schlachtfeld entscheiden ist, sondern der rechte Flügel allein den Sieg bringen kann.

Wenn sich also die bisher getrennt marschierenden 68000-Kämpfer dazu entschließen könnten, gemeinsam — so utopisch das auch klingen mag — mit vereinten Kräften dem eigentlich richtigen System(en) zum Sieg zu verhelfen, so wäre aus meiner Sicht dann der Vernunft Genüge getan. Nicht aber mit diesen Zwistigkeiten unter Brüdern ist das Ziel zu erreichen. Alle haben auf das richtige Pferd gesetzt, sie müßten sich nur einig werden über eine einheitliche Sattelnorm.

Diese Vision eines geschlossenen Marschierens der 68000er-Phalanx mag heute noch wie Zukunftsmusik klingen. Aber irgendwann ist die Zeit gekommen, in der die PS/2-Computer mit mehr Farben und benutzerfreundlicheren Oberflächen die eigentliche Domäne der Amigas, Macs und Ataris bedroht. Dann muß sich beweisen, wie widerstandsfähig die »andere« Computerszene ist.

Aber auch wenn keine Einigung zwischen den drei großen 68000er-Sympathisanten zustande kommt, der Amiga wird seinen Weg problemlos alleine gehen. Die verkauften Stückzahlen haben mittlerweile eine Schwelle überschritten, wo es eigentlich kein Zurück mehr gibt. Der C 64 hat es vorexerziert. Der Amiga kann es als einziger nachahmen, ein Millionenseller in Deutschland zu werden.

Herzlich Ihr
Albert Absmeier
Chefredakteur

A. Absmeier



In einem Gespräch mit Jay Miner, dem Entwickler des Amiga 1000, erfahren Sie, warum der Mitgründer der Firma Amiga die Behauptung aufstellt: »Das ist nicht mehr mein Amiga.« Doch damit nicht genug. Jay Miner findet den Amiga völlig veraltet und übt herbe Kritik an Commodore.

Die Benutzeroberflächen sind nicht nur bei den Heimcomputer-Herstellern ein heißes Thema. In der **Ausgabe 2/88** der 68000er lesen Sie, welche Meinung Joy Mountford von Apple Computers vertritt und wie sie die Zukunft der Benutzeroberflächen sieht.

Ein neues Textverarbeitungsprogramm für den Amiga macht von sich reden. Mit DesignText tritt der kanadische Software-Markt den Konkurrenten gegenüber. Eine hochauflösende Druckerausgabe und die mehrspaltige Textdarstellung sprechen sicherlich für sich. Lesen Sie, mit welchen Vorzügen DesignText noch ausgestattet ist.

Der zweite Teil des Kurses über die Amiga-Spezialchips führt Sie in die Interruptstruktur und den Aufbau eines Playfields ein. Ein Copperprogramm, in dem Sie alles Gelernte in die Praxis umsetzen können, rundet diesen Teil des Spezialchip-Kurses ab.

Ist der Sound des Amiga nicht faszinierend? Bis jetzt mußten Sie, wenn Sie die Amiga-Klänge in perfekter Qualität genießen wollten, den Amiga mit der Stereoanlage verbinden. Mit unserer Hardware-Bastelei, dem Audio-Verstärker, ist endlich Schluß mit dem Kabelsalat.

In einem ausführlichen Plottertest stellen wir Ihnen den HPX-84 vor. Versehen mit einer besseren Mechanik und neuer Steuersoftware tritt der Selbstbau-Plotter den Kampf gegen seine teuren Kollegen an.



In der **Ausgabe 2/88** gibt es das Denkspiel Quadromania, das Sie stundenlang vor den Amiga fesseln wird. Wenn Sie Ihren Amiga selbst programmieren wollen, hilft Ihnen unser Basic-Kurs. Einfache Beispiele zeigen anschaulich, wie man durch simple Routinen viel Wirkung erhält. Wenn Sie gerade an den Sommer denken, lesen Sie in Happy-Computer alles über Computercamps. Wo gibt es welche, und für welchen Preis? Planen Sie jetzt Ihren Sommerurlaub. Wie geht es weiter mit Btx? Postminister Schwarz-Schilling steht Rede und Antwort.

In der **Ausgabe 3/88** dreht sich alles um Ihre High-Tech-Karriere. Welche Berufe gibt es für Computerinteressierte? Welche Zukunftsaussichten haben die verschiedenen Berufe? Wie bewerben Sie sich richtig? Worauf Sie achten sollten, finden Sie ebenso wie einen großen Stellenmarkt. Unerläßlich ist natürlich Erfahrung am Computer. Im Amiga-Basic-Kurs erfahren Sie, wie wenige Zeilen ein Programm benutzerfreundlicher machen.

Emulatoren sind die Verbindung zweier Welten. Ein Computer simuliert einen anderen. Doch halten die Emulatoren, was sie versprechen? Wir sind diesen Fragen auf den Grund gegangen. Emotionsgeladen sind alle Diskussionen über Hacker, besonders nach dem Coup im NASA-Netz. Doch wer sind die deutschen Hacker? Sind sie moderne Robin Hoods oder gewöhnliche Kriminelle? Was sind ihre Ideale, ihre Ideen, ihre Ziele und ihre Geschichten?

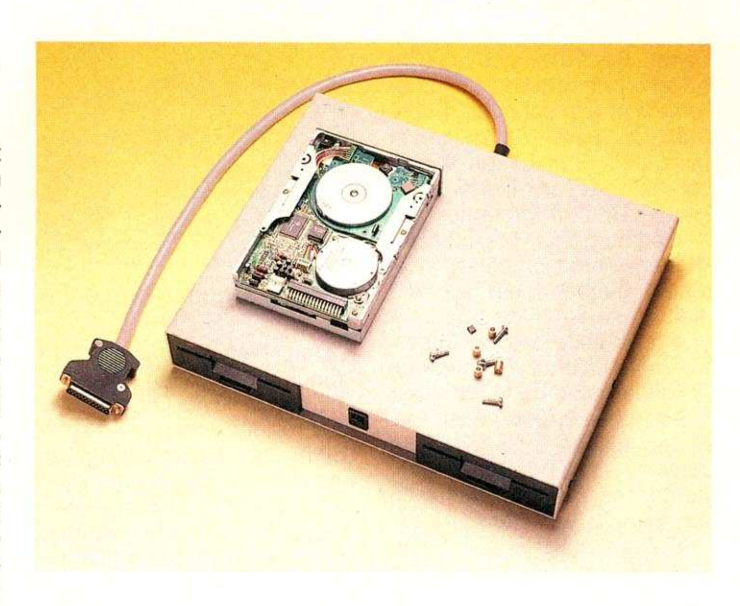
Durch den Spiele-Sonderteil sind Sie immer über die neuesten Spiele für den Amiga informiert. In dieser Ausgabe testen wir unter anderem das Geschicklichkeitsspiel Tetris und das Knobelspiel Power Play.

News im Überblick

Kompatible Ausweichstationen

Zu den vielen auf dem Markt erhältlichen Zusatzlaufwerken für die Amiga-Modelle gesellen sich nun zwei weitere Diskettenstationen. Zum einen ein NEC-Laufwerk (im Bild oben), welches zum internen Einbau in den Amiga 2000 gedacht ist und später als »df1« angesprochen werden kann. Der Preis für das Einbaulaufwerk: 229 Mark. Die zweite neue Diskettenstation ist ein aus zwei NEC 1035-Laufwerken aufgebautes Doppellaufwerk (im Bild unten) zum externen Anschluß an jedes Amiga-Modell. Das Doppellaufwerk ist zum Preis von 649 Mark erhältlich. (dm)

Datentechnik Bittendorf, Postfach 100248, 6360 Friedberg, Tel. 06031/61950 ▶



Das kleine Einzellaufwerk eignet sich zum Einbau in den A2000, das untere Doppellaufwerk in jeden Amiga

Info für Programmierer

Wichtig für alle Software-Hersteller, die Produkte vertreiben wollen, auf denen Workbench-Files enthalten sind: Die Workbench-Files (CLI und ähnliches) sind nicht Public Domain. Die aktuelle Workbench-Lizenz besteht aus einem 5-Jahres-Vertrag. Gezahlt wird jährlich in 100-Dollar-Beträgen. Eine Lizenz für die Verwendung der Include-Files ist für eine einmalige Gebühr von 50 Dollar über die Dauer von fünf Jahren gültig. Das gab Commodore International in einem Rundbrief für US-Entwickler bekannt.

(M. Kohlen/jk)

Commodore Inc., Attn. Lauren Brown, 1200 Wilson Drive, Westchester, PA 19380, USA

U-Boote einmal anders

Kaum hat es den Zwischenfall mit einem russischen U-Boot in den Gewässern der Ostsee gegeben, hat sich ein findiger Autor an die Arbeit gemacht und aus dem ganzen Vorfall einen Roman geschrieben. Nach diesem Bestseller »The Hunt for Red October« von Tom Clancy ist das gleichnamige Spiel für den Amiga entstanden. Dabei übernehmen Sie allerdings die Rolle des Kapitäns des russischen Atom-U-Boots »Roter Oktober«. Prompt ist natürlich die ganze Welt hinter dem Geheimnis des neuen superleisen Antriebssystems her. Spannung ist also allemal gesichert, wenn man mit dem gut menügesteuerten Spiel versucht, den Feinden zu entkommen. Hoffen kann man nur, daß diese U-Boot-Simulation nicht den gleichen Weg wie ein ähnliches Spiel nimmt, das immer noch zu den besten dieser Gattung gehört, jedoch leider indiziert wurde. (jk)

Ariolasoft, Postfach 1350, 4830 Gütersloh 1
Preis: 75 Mark



Auf der Jagd nach dem supermodernen U-Boot

Neuer Digitizer: Perfekt Vision

Einen Video-Digitizer, der den H.A.M.-Modus unterstützt, bietet SunRize Industries USA an. Mit dem laut Mitteilung des Herstellers in Echtzeit arbeitenden Digitizer können Sie Bilder von einem VCR, Camcorder oder Videorecorder in jeden Amiga einspeisen. Die Zeit für die Erfassung eines Bildes soll nur 1/60 Sekunde betragen. »Perfekt Vision« digitalisiert jede Aufnahme in 16 Graustufen. Der Digitizer wird für den IBM-XT/AT und den Amiga 500/1000/2000 angeboten. Die Amiga-Version unterstützt den 320 x 200- und 320 x 400-H.A.M.-Modus. Mit Hilfe von Farbscheiben können Sie Aufnahmen in bis zu 32 Farben digitalisieren. Alle Aufnahmen können im IFF gespeichert werden und mit jedem IFF-kompatiblen Mal- und DTP-Programm, zum Beispiel Deluxe Paint II, verarbeitet werden. Der Preis von Perfekt Vision in den Vereinigten Staaten beträgt zur Zeit 219,95 Dollar. (ub)

Anthony Wood, SunRize Industries, 3801 Old College Road, Bryan, Texas 77801, (409) 846-1311

Commodore

Der Geschäftsführer der deutschen Commodore-Gesellschaft, Winfried Hoffmann, wurde in das Europa Management berufen. Seit 1.1.88 unterstehen ihm die gesamten Vertriebs- und Marketing-Aktivitäten der zwölf europäischen Commodore-Gesellschaften.

Sein Nachfolger in der Geschäftsführung wird Heinz Wiening (48), zuletzt für Philips tätig.

Commodore-Computer waren im Weihnachtsgeschäft 1987 der Renner. 8500 Amiga 2000 wurden verkauft. Sensationell verkaufte sich der Amiga 500, der wohl als neuer Standard für Heimcomputer angesehen werden kann, mit fast 50000 Stück. Erfolgreicher war nur der C 64 mit über 60000 abgesetzten Exemplaren. Bei den MS-DOS-Geräten setzte Commodore die Rekordzahl von 25000 ab. Derzeit hält Commodore nach eigenen Angaben bei Heimcomputern einen Marktanteil von 85 Prozent. (aa)

Erweiterte Textdimensionen

In letzter Minute erreichte uns die erste Version der neuen Textverarbeitung »Design Text«. Dieses neue Programm wartet mit Funktionen auf, die bisher auf dem Amiga nicht realisiert wurden und scheint eine ernsthafte Gefahr für alle Mitkonkurrenten aus dem heiß umkämpften Textverarbeitungslager zu sein. Laut Hersteller soll sich auch auf 9-Nadel-Druckern eine gute Druckqualität erzielen lassen. Verschiedene Zeichensätze, eingebaute Datenbank, Mailmerge, Einbindung von Grafiken, Multiwindowing und vieles mehr sollen mit Design Text zum Kinderspiel werden.

Wir haben bereits die Version 0.95 zum Antesten erhalten. Die endgültige Version soll auf der AmiExpo im Januar in Los Angeles vorgestellt werden. Texte können durch einfache Anwahl des voll Intuition-unterstützten Programmes in mehreren Spalten formatiert werden, so daß sich mit IFF-Grafiken Desktop Publishing ähnliche Ausdrucke bearbeiten lassen. Dabei werden mit den besonderen Zeichensätzen sowohl mathematische Sonderzeichen als auch kyrillische Buchstaben einwandfrei unterstützt und ausgedruckt.

Weiterhin sind Sie in der Lage, äußerst komfortabel Kopf- und Fußzeilen sowie Seitenzahlen (und deren Position) in den Text einzubringen. Auch das Anlegen von Inhaltsverzeichnissen und Verweisen mit Hilfe von Indizes beziehungsweise Fußnoten erweist sich als einfach und gut durchdacht. Beim Anlegen des Inhaltsverzeichnisses hilft unter anderem die schnelle Such-Funktion, deren Arbeitsgebiet mit vielen Funktionen eingeschränkt oder erweitert werden kann. Unterprogramme zur Versendung von »Mailmerge«-Briefen, Rundschreiben also, fehlen ebensowenig wie die gleichzeitige Verarbeitung mehrerer Texte und das Importieren von Daten aus anderen Textverarbeitungen.

Das über 90 000 Worte fassende, bisher noch englische Lexikon, hilft natürlich dem deutschen Benutzer recht wenig. Ebenso verhält es sich mit der automatischen Trennfunktion, die bei deutschen Texten meist fehlerhaft trennt.

Design Text ist ein vielversprechendes neues Programm auf dem Amiga-Markt. Lesen Sie in einer unserer nächsten Ausgaben den ausführlichen Testbericht.

(Ottmar Röhrig/jk)

Compustore, Fritz-Reuter-Str. 6, 6000 Frankfurt, Tel. 069/567399
Geplanter Preis: 328 Mark

Zukünftiges Ballern

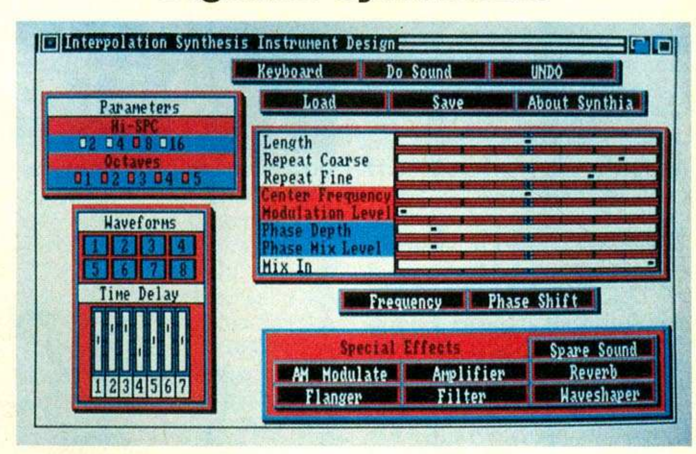
Mit »Emetic Skimmer« ist wieder ein Spiel deutscher Programmierer veröffentlicht worden (Bild). Mit dem Raumfahrzeug Emetic Skimmer geht es dabei in ferner Zukunft um die Rettung des ganzen Sonnensystems. Ein gefährlicher Computer mit Künstlicher Intelligenz, der sich von seiner Basis auf Pluto dazu anschickt, die Macht zu übernehmen, muß in vielen Einzelmissionen zerstört werden. Grafik und Sound sind ansprechend entworfen, nur das Spielprinzip kommt einem irgendwie bekannt vor, ist es doch angelehnt an das schon vom C 64 bekannte »Survivor«. (jk)

Kingsoft, Grüner Weg 29, 5100 Aachen 1, Tel. 0241/152051, Preis: 49,95 Mark



Das Raumfahrzeug Emetic Skimmer im Kampf

Digitaler Synthesizer



Digitaler Synthesizer mit Komfort

Den bisher einzigen vollständigen digitalen Synthesizer haben »The Other Guys« jetzt für den Amiga vorgestellt (Bild). Damit können Sounds und Instrumente für nahezu alle am Markt befindlichen Musikprogramme erstellt werden. In mehreren Programmodulen sind Parameter für Filter, Hüllkurve, Flanging, Hall und anderes einstellbar. Bestimmte Arten von Instrumenten wie etwa Trommeln oder Saiteninstrumente werden mit speziellen Funktionen wie additive, subtraktive oder interpolative Synthese leichter beeinflußbar. Der Clou von Synthia liegt in der Effekt-Sektion, die etwa eine Änderung von Tremolo und Verstärkung auch digitalisierter Sounds zuläßt. Ein ausführlicher Test folgt in der nächsten AMIGA. (jk)

Atlantis, Dunantstr. 53, 5030 Hürth
Preis: 248 Mark

Video-Digitizer aus England

G.T.I. hat den Vertrieb eines von JCL Business Systems in England produzierten Video-Digitizer übernommen. »Digipic« kann nach den Angaben der englischen Bedienungsanleitung an jeden Amiga über den Parallel-Port angeschlossen werden. Ihr Amiga benötigt mindestens 512 KByte, jedoch für ein vernünftiges Arbeiten sind 1 MByte wünschenswert. Als Bildquelle benötigen Sie eine Kamera mit Monochrom-Ausgang. Zur besseren Überwachung der Kameraeinstellung können Sie optional einen zweiten Monitor direkt am Digitizer anschließen. Laut Aussage des Herstellers hat sich ein Commodore 1701 bewährt. Der Digitizer soll in der Lage sein, ein Bild in Echtzeit ($^1/_{50}$ Sekunde) zu digitalisieren. Das Bildformat beträgt standardmäßig 320 x 200 Punkte. Die Darstellung erfolgt in 16 Graustufen. Mit einem zusätzlichen »erweiterten Graustufen-Adapter« können 32 Grautöne verwendet werden. Jedes Farbbild muß mit den schon von anderen Systemen her bekannten Farbscheiben dreimal digitalisiert werden. Die Software erzeugt aus den Signalen eine Abbildung mit 32 Farben. Weitere Funktionen der Software sind:

Die Digitalisierung wird durch Mausklick gestartet.

Sie können jede Aufnahme im ILBM-Format (nach Wunsch komprimiert) oder im intern verwendeten Digitizerformat speichern.

Sie haben ebenfalls die Option, nur einen Ausschnitt des Bildschirms zu speichern.

Der »Loop-Modus« dient zum kontinuierlichen Digitalisieren und Darstellen eines aufzunehmenden Bildes.

Im »Multi-Modus« werden mehrere Aufnahmen in unmittelbarer Folge aufgenommen.

Jede Aufnahme wird zunächst im 64-KByte-Speicher des Digitizers festgehalten und dann erst an den Amiga übertragen. Bei Bedarf können Sie ein bereits geschossenes Foto mit »Re-Read« ein zweites Mal einlesen.

Die genauen Features des Digitizers werden wir Ihnen in einer unserer nächsten Ausgaben vorstellen. Digipic ist in Deutschland direkt von G.T.I. zu einem Verkaufspreis von 1185 Mark zu beziehen. (ub)

GTI GmbH, Oberhöchstadter Str. 53b, 6370 Oberursel

PAL-Modus jetzt unterstützt

In der neuesten Version des in Ausgabe 11/87 getesteten Programmes »Sculpt-3D« wird nun auch die europäische PAL-Auflösung bei der Berechnung von Bildern voll unterstützt. Das Programm bietet jetzt neben der eigentlichen Berechnung von Bildern mit bis zu 640 x 512 Punkten auch noch die Möglichkeit des vertikalen und horizontalen Overscans. Damit verschwinden sämtliche Ränder aus den Bildern und es wird ein volles Fernsehbild erzeugt.

Eine weitere wichtige Neuerung ist bei den eigentlichen Ray-Tracing-Routinen zu bemerken. So werden jetzt Objekte mit den Attributen »Glas« oder »Spiegel« mehr als dreimal schneller berechnet, als das noch bei der ersten Version der Fall war. Um dies zu erreichen, mußte die Zeit zur Vorberechnung des Bildes (die Zeit, in der sich der Mauszeiger in zwei Zahnräder verwandelt) zwar etwas erhöht werden, aber der Lohn dafür ist um so größer, da auch normale Bilder schneller berechnet werden. Durch die Implementation des PAL-Modus gehören auch die fehlerhaften Bildspeicherungen nun der Vergangenheit an. Auch der Overscan wird jetzt problemlos verarbeitet.

Ein neuer Menüpunkt kann nun auch den »Verwischungsgrad« (Dither) zwischen zwei Farben bestimmen. Eine aus Deluxe-Paint II bekannte Funktion, die interessante Effekte beinhaltet. Weiterhin wurden noch einige Demodateien beigelegt, die wieder faszinierende Objekte zeigen. Die Berechnung dieser Bilder kann aber, wie gehabt, nur auf einem Amiga mit mindestens einem MByte RAM erfolgen.

Ein Nachteil des PAL-Modus soll hier aber auch nicht verschwiegen werden. Konnten Sie bisher einigermaßen große Objekte auf einem Amiga mit 512 KByte RAM erzeugen, so sollten Sie sich beim Kauf des neuen Sculpt-3D doch gleich eine Speichererweiterung mitbestellen. Durch den wesentlich höheren Speicherverbrauch von PAL-Bildern gegenüber NTSC-Darstellungen sind Sie mit nur 512 KByte RAM doch sehr in Ihrer Objektauswahl eingeschränkt. Ein Trick hierzu: Geben Sie im Info-Menü des Sculpt-Hauptprogrammes »NTSC« ein, werden alle Bilder wie früher berechnet. Die Umkehrung dieser Funktion ist für deutsche Benutzer nicht interessant, ermöglicht sie doch den amerikanischen Usern die Erzeugung von PAL-Bildern auf ihren NTSC-Amigas.

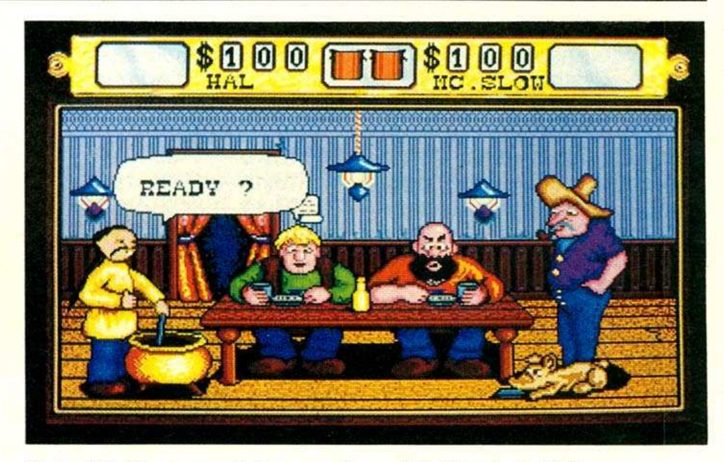
Alles in allem ist die neueste Version von Sculpt-3D in sehr vielen Punkten verbessert worden und bleibt damit an der Spitze der Video-/Grafik-Programme. (Ottmar Röhrig/jk)

Preis: zirka 190 Mark in gutsortiertem Fach- und Versandhandel

Ab in den Westen

Eine andere Art von Olympischen Spielen gibt es jetzt mit »Western Games« für den Amiga (Bild). Darin dürfen Sie zu zweit oder allein gegen den Computer in Disziplinen antreten, die alle ihren Ursprung im »Wilden Westen« haben. Obwohl das Spiel im wesentlichen in die Gattung der Sportspiele einzuordnen ist, sind einige Lacher bestimmt garantiert. Denn so bierernst geht es natürlich im Wilden Westen nicht zu. Insgesamt sechs Disziplinen versprechen viel gute Laune, vor allem, wenn man zu zweit spielt: Armdrücken, Schießen, Spucken, Tanzen, Melken und Essen. Die Grafik ist nett gemacht und an vielen Stellen zusätzlich animiert. Einige Gags wie etwa Schienbeintritte für den Gegner beim Wettessen lockern das Ganze erheblich auf. Natürlich gibt es auch eine auf Diskette gespeicherte High-Score-Liste. (jk)

Ariolasoft, Postfach 1350, 4830 Gütersloh 1
Preis: 60 Mark



Beim Wettessen gibt es auch mal Schienbeintritte

FlashEd ist da

So mancher Amiga-Programmierer hat sich sicher schon über das Fehlen eines brauchbaren Programmeditors auf seinem Computer geärgert. Ed und Edit, auf der Workbench-Diskette mitgeliefert, lassen einiges an Komfort vermissen. Beim neuen »FlashEd« handelt es sich um einen vollständig in Assembler geschriebenen Programmeditor. Auf unnötige Spielereien wurde zugunsten der Übersichtlichkeit und Bedienung verzichtet.

Für einen ersten Kurztest lag uns die Version 1.2 exklusiv vor. FlashEd ist ein Programmeditor, der auf dem vollen PAL-Bildschirm des Amiga 29 Zeilen mit 80 Zeichen darstellt. Eine Statuszeile an der Bildunterkante informiert über die momentane Cursorposition in Zeile und Spalte sowie über den noch freien Textspeicherraum. Auf einen Kopierschutz wurde lobenswerterweise verzichtet. Zunächst einmal fällt auf, daß FlashEd sowohl über Menüleisten als auch vollständig über die Tastatur bedienbar ist. FlashEd ist standardmäßig für Texte bis 40 KByte Länge ausgelegt. Wer längere Texte bearbeiten will, kann beim Laden des Programms durch Angabe eines Parameters beliebige Textlängen einstellen. Beschränkt ist man hier nur durch den hardwaremäßig verfügbaren Speicher. Natürlich ist es auch möglich, die Multitasking-Fähigkeiten des Amiga weiterhin auszunutzen. FlashEd arbeitet mit einem Pufferspeicher, der entweder extern aus anderen Dateien oder mit einem Block aus der gerade bearbeiteten Datei geladen werden kann.

Das Werkzeugmenü enthält zwei Suchfunktionen, vorwärts und rückwärts. Bei beiden Funktionen wird von der momentanen Cursorposition aus mit der Suche nach der in einer Statuszeile am unteren Bildschirmrand eingegebenen Zeichenfolge begonnen. Bei allen Funktionen, die den geladenen Text in irgendeiner Weise löschen würden, wird eine Sicherheitsabfrage durchgeführt. Einige weitere, sehr nützliche Funktionen sind über die Tastatur ansprechbar. So ist es möglich, eine Zeile rechts oder links vom Cursor zu löschen. Das schnelle Scrollen und Blättern, das Programmänderungen auch in großen Programmen zur Freude macht, zeichnet den FlashEd aus. Die Funktionstasten können jeweils mit einer maximal 32 Zeichen langen Zeichenfolge belegt werden, so daß dem Programmierer insgesamt 20 auf Tastendruck einsetzbare Standardfloskeln zur Verfügung stehen. Eine Besonderheit ist auch die Möglichkeit, Sonderzeichen über ihren ASCII-Code einzugeben.

Zum Durchblättern eines 100 KByte langen Textes (zirka 2000 Zeilen mit 80 Zeichen) benötigt man nur etwa 10 Sekunden.

Programme dieser Art zeigen, daß der Amiga nicht nur eine interessante Spielmaschine ist, sondern auch im professionellen Bereich stehen kann. Wer noch einen deutschsprachigen Editor sucht, sollte sich FlashEd einmal ansehen. (Jochen Ewald/jk)

C. Klimm, Am Alten Feld 16a, 5270 Gummersbach 31, Preis: 59,90 Mark

Bewegte Titel

Die GoAmiga!-Serie von Softwareland hat sich um ein neues Produkt erweitert. Ab jetzt sind mit »GoAmiga! Titel« verschiedene Laufschriften und digitalisierte Toneffekte mit IFF-Bildern kombinierbar. Im Programm können Sie mit Menü- und Maussteuerung Ihre eigenen Titel und Videovorspänne zusammenstellen. Alle Zeichensätze mit bis zu 80 x 80 Punkten Größe dürfen verwendet werden. (jk)

Softwareland, Franklinstr. 27, CH-8050 Zürich, Tel. 0041-1-3115959
Preis: 59 SFr.

Speicherriese



40-MByte-Festplatte mit schneller Zugriffszeit

Für den Amiga 1000 ist eine weitere Festplatte erhältlich, die Speicherplatzprobleme lösen soll. Für die schnelle Hard-Disk, die mit verschiedenen Kapazitäten angeboten wird, gewährt der Hersteller ein halbes Jahr Garantie. Die Festplatte besitzt eine eingebaute Stromversorgung, so daß der Amiga 1000 nicht belastet wird. Das Gerät wird mit der entsprechenden Installationssoftware und einer Demodiskette ausgeliefert. Für die verschiedenen Kapazitäten sind folgende Preise angegeben:

20 MByte — 1498 Mark
40 MByte — 1998 Mark
72 MByte — 3498 Mark

Ein Test der 40-MByte-Platte erfolgt in einer der nächsten Ausgaben der AMIGA. (dm)

Frank Elektronik GmbH, Matthiasstr. 3, 8500 Nürnberg 80, Tel. 0911/327732-327791

Wenn der Kleine mit dem Großen

Die Programmeingabe und -entwicklung auf einem Sharp-PC gestaltet sich durch die winzigen Tasten recht schwierig. Da bietet es sich den Amiga-Besitzern doch geradezu an, mit einem guten Editor die Programme auf ihrem Computer zu entwerfen, zum Sharp zu überspielen und sie auf die Funktionsfähigkeit zu überprüfen. Genau diese Aufgaben übernimmt das »Transfile«-Paket. Neben Programmen können auch Speicherbereiche in der von Sharp bekannten hexadezimalen Darstellungsweise sowie Variablen oder Variablenfelder übertragen werden. Dabei kann der Amiga sowohl Sender als auch Empfänger sein. Es steht somit die Diskettenstation als Speichermedium für Sharp-Programme zur Verfügung.

Die Transfile-Software unterscheidet bei Programmen streng zwischen der Binär- und der ASCII-Version. In der Binärversion können Programme direkt zum Sharp übertragen und empfangen werden. Doch editierbar sind sie, durch die speichersparende Verwendung von Basic-Tokens im Sharp, so nicht. Dazu müssen diese Daten erst mit Hilfe der mitgelieferten Tokentabellen in das ASCII-Format umgewandelt werden. Die fehlerlose Umwandlung erfolgt automatisch. Der einzige Unterschied eines solchen Programmes gegenüber seinem Original besteht in der Darstellung von Zeichen, die der Amiga nicht beherrscht. Dies ist zum Beispiel das Wurzelzeichen. Solche Codes werden dann als Hexadezimalzahl in geschweiften Klammern angezeigt und bei der Rückwandlung in das Binärformat auch richtig erkannt.

Damit auch bei anderen Sharp-Rechnern keine Probleme auftauchen, wird Transfile gleich mit allen zur Zeit erhältlichen Tokentabellen ausgeliefert. Die Übertragung selbst gestaltet sich nach Herstellerangaben einfach, da beim Sharp nur die Standardbefehle CLOAD und CSAVE mit eventuell Parametern und beim Amiga nur zwei Mausklicks nötig sind. Zeit muß man natürlich ebenfalls aufwenden, da das Interface der Sharprechner nur maximal 300 Bit/s überträgt. (Ottmar Röhrig/dm)

Yellow-Computing, Im Weingarten 21, 7101 Hardthausen.
Preis: 99 Mark

Jeder sein eigener Boß

Auf dem Spielesektor gibt es für den Amiga noch einige Lücken bei bestimmten Spielearten. Von Fastworks kommt jetzt ab Januar mit »Manager« ein Wirtschaftsspiel auf den Markt für alle, die selbst einmal ihr eigener Boß sein möchten. Maximal 20 einzelne Spieler können sich am Kampf um das Überleben von Firmen und Konzernen beteiligen. Trotz vieler Parameter hat das Spiel nicht den Anspruch einer kompletten Wirtschaftssimulation. Das Ziel ist es, Manager zu werden und nicht pleite zu gehen. Dazu wird vom Computer ein Marktgeschehen mit Angebot und Nachfrage, Aktien, Versicherungen und sogar Konkursen simuliert. Durch Einstellung der spielbestimmenden Parameter, wie etwa Kurse und Zinsen, kann man je nach Bedarf den Schwierigkeitsgrad erhöhen. Das Programm wird komplett in Deutsch zum Preis von 49,50 Mark ausgeliefert. (jk)

Fastworks, Postfach 100449, 5090 Leverkusen 1

Impressum

Herausgeber: Carl-Franz von Quadt, Otmar Weber

Geschäftsführender Chefredakteur: Michael Scharfenberger

Chefredakteur: Albert Absmeier (aa)
Redaktion: Peter Aurich (pa), René Beaupoil (rb), Ulrich Brieden (ub), Jörg Kähler (jk), Dieter Mayer (dm)

Redaktionsassistenz: Andrea Kaltenhauser (202)

Fotografie: Jens Jancke, Claudia Kränzle

Titelgestaltung: Heinz Rauner, Grafik Design, Werner Nienstedt

Layout: Friedemann Porscha (Ltg.), Dagmar Berninger, Willi Gründl

Auslandsrepräsentation:
Schweiz: Markt & Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042-415656, Telex: 862329 mut ch
USA: M & T Publishing, Inc: 501 Galveston Drive, Redwood City, CA 94063; Tel. (415) 366-3600, Telex 752-351

Manuskripteinsendungen: Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten werden, so muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt & Technik Verlag AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Mit der Einsendung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt & Technik Verlag AG verlegten Publikationen und dazu, daß Markt & Technik Verlag Geräte und Bauteile nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben läßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Produktionsleitung: Klaus Buck

Gesamtanzeigen-Verkaufsleitung: Ralph Peter Rauchfuss (126)

Anzeigenverkaufsleitung: Alexander Narings (780)

Anzeigenleitung: Alicia Clees (313)

Anzeigenverkauf: Christine Pfäffinger (781)

Anzeigenverwaltung und Disposition: Patricia Schiede (172), Monika Burseg (147)

Anzeigenformate: 1/1-Seite ist 266 Millimeter hoch und 185 Millimeter breit (3 Spalten à 58 mm oder 4 Spalten à 43 Millimeter). Vollformat 297 x 210 Millimeter. Beilagen und Beihefter siehe Anzeigenpreisliste.

Anzeigenpreise: Es gilt die Anzeigenpreisliste Nr. 2 vom 1. Januar 1988
Anzeigengrundpreise: 1/1 Seite sw: DM 4900,—. Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala je DM 800,—. Vierfarbzuschlag DM 2200,—.
Anzeigen in der Fundgrube: Private Kleinanzeigen mit maximal 4 Zeilen Text DM 5,— je Anzeige.
Gewerbliche Kleinanzeigen: DM 12,— je Zeile Text.
Auf alle Anzeigenpreise wird die gesetzliche MwSt. jeweils zugerechnet.

Anzeigen-Auslandsvertretungen:
England: F. A. Smyth & Associates Limited 23a, Aylmer Parade, London, N2 OPQ, Telefon: 0044/1/3405058, Telefax: 0044/1/3419602

Taiwan: Third Wave Publishing Corp. 1 — 4 Fl. 977 Min Shen E. Road, Taipei 10581, Taiwan, R.O.C., Telefon: 00886/2/7630052, Telefax: 00886/2/7658767, Telex: 078529335

Vertriebsleiter: Helmut Grünfeldt (189)

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebsgesellschaft mbH, Hauptstätterstraße 96, 7000 Stuttgart 1, Telefon (0711) 6483-0

Bezugsmöglichkeiten: Leser-Service: Telefon 089/4613-249. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen. Das Abonnement kann jederzeit zum Ende des bezahlten Zeitraums gekündigt werden.

Bezugspreise: Das Einzelheft kostet DM 7,—. Der Abonnementspreis beträgt im Inland DM 79,— pro Jahr für 12 Ausgaben. Darin enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren. Der Abonnementspreis erhöht sich um DM 18,— für die Zustellung im Ausland, für die Luftpostzustellung in Ländergruppe 1 (z. B. USA) um DM 38,—, in Ländergruppe 2 (z. B. Hongkong) um DM 50,—, in Ländergruppe 3 (z. B. Australien) um DM 68,—.

Druck: R. Oldenbourg GmbH, Hürderstr. 4, 8011 Kirchheim

Verantwortlich: Für redaktionellen Teil: Albert Absmeier. Für Anzeigen: Alicia Clees.

Redaktions-Direktor: Michael M. Pauly

Vorstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt & Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon 089/4613-0, Telex 522052

Telefon-Durchwahl im Verlag:
Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089-4613 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.

Aktionäre, die mehr als 25% des Kapitals haben: Otmar Weber, Ingenieur, München, Carl-Franz von Quadt, Betriebswirt, München;
Aufsichtsrat: Dr. Robert Dissmann (Vorsitzender), Karl-Heinz Fanselow, Eduard Heilmayr

Basic-Cursor soll größer werden

Wer kennt einen Weg, um den Cursor des Amiga-Basic im List-Fenster besser sichtbar zu machen? Ein breiterer Strich wäre sehr nützlich.

HANS-JOACHIM SCHMIDT

Zwei Amiga koppeln

Ist es denkbar, daß ein Amiga 1000 auf die Laufwerke und eine Hard-Disk eines Amiga 2000 und umgekehrt zugreifen kann und vielleicht sogar Daten austauscht?

An welchem Port ist eine Kopplung am einfachsten?

Ist eine reine Software-Lösung realisierbar, oder ist eine Netzwerkkarte nötig?

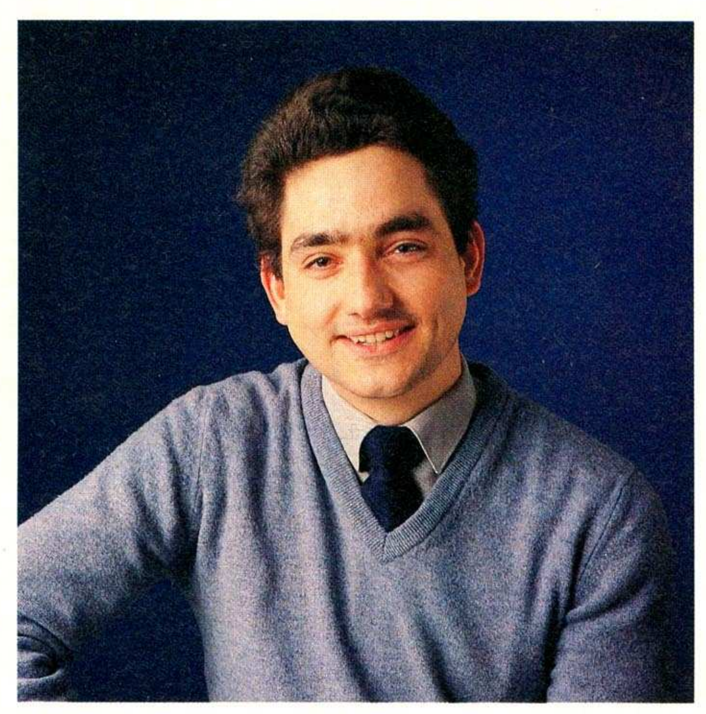
MICHAEL SCHARZ

Eine Netzwerkkarte ist für Ihr Projekt auf jeden Fall die beste Lösung. Nur dann können Sie ungehindert auf die Laufwerke eines anderen Computers zugreifen.

Eine reine Softwarelösung wäre denkbar. Allerdings müßten die angeschlossenen Computer dann laufend kommunizieren und überprüfen, ob einer der anderen Computer etwas möchte. Über Modem beziehungsweise die serielle Schnittstelle könnten die Computer Daten austauschen. Auch die Parallel-Ports können programmiert werden, um Daten zwischen zwei Amiga zu transportieren. Ein interessantes Projekt: Jeder der verbundenen Amiga müßte Daten lesen können. An einem Code muß er erkennen, daß die Daten für ihn sind. Handelt es sich um eine Nachfrage eines anderen Amiga, müßte der angesprochene Amiga reagieren. Zum Beispiel indem er ein File von Diskette an den rufenden Amiga überträgt. Wer dies programmiert, muß sich gut mit der Hard- und Software auskennen. (ub)

FRAGEN SIE

Wenn Sie Probleme mit dem Amiga, mit Peripheriegeräten oder mit der Software haben, stellen Sie Ihre Fragen ans Leserforum des AMIGA-Magazins. So können Sie mit anderen Lesern in Kontakt treten, die bereits eine Lösung gefunden haben.



Fragen und antworten Sie im Leserforum

Hilfe zu LPRINT

In Ausgabe 12/87 ist auf Seite 20 des Leserforums eine Anfrage von Dr. T. Reichelt abgedruckt. Als ich meinen Drucker (NEC P6) bekam, konnte ich ebenfalls nicht mit LPRINT und LLIST arbeiten. Die Lösung: Der Interpreter greift bei diesen Befehlen auf das »printer-device« zu. Dieses verwendet den Druckertreiber und bei entsprechender Einstellung in den Preferences das Parallel-device. Damit prt: seine Aufgabe erfüllen kann, darf par: nicht geöffnet sein. Durchforsten Sie Ihr Programm, ob vor der Verwendung von LPRINT das Parallel-Device geöffnet wird. Falls dies nicht hilft, so müssen Sie das Parallel-Device, das Printer.device und den Druckertreiber auf Ihrer Startdiskette überprüfen.

OLAF ZERRES

Fremdmonitore am Amiga

Ich habe vor, mir einen Amiga 2000 zu kaufen. Da ich schon einen RGB-Monitor besitze (Modell CCM-1480 von Orion), möchte ich gerne wissen, ob ich diesen am Amiga verwenden kann.

MARCEL WÜRZBURG

Ausgabe 10/87

Ich habe mir einen Amiga gekauft und ähnliche Schwierigkeiten bekommen. Doch es ist möglich, den Monitor (Orion CCM-1480) anzuschließen. Die Qualität ist zwar nicht so gut wie die des Originalmonitors, aber auch nicht zu verachten. Ein Verbindungskabel muß sich jeder selber basteln, da es noch keines zu kaufen gibt. So lautet auch die Auskunft des Herstellers von Computerkabeln Lindy. Bastler benötigen diese Einzelteile:

— einen 6poligen Diodenstecker
— einen 23poligen Computer(Video)-Gegenstecker
— und ein 5adriges Kabel mit Abschirmung

Folgende Anschlußbelegung ist erforderlich:

vom Amiga 2000 zum Monitorstecker	(RGB)
3-(Analog Rot)	1
4-(Analog Grün)	2
5-(Analog Blau)	3
11-(Horizontal Sync)	4
13-(GND) Abschirmung	5
12-(Vertical Sync)	6

JÖRG LÖFFLER

(Anmerkung der Redaktion: Sollten beim Anschluß mit einem selbstgebauten Kabel Schäden an Ihren Geräten entstehen, kann keine Haftung übernommen werden)

True-Basic

True-Basic läuft sowohl auf dem Amiga als auch auf MS-DOS-Computern. Der Wortschatz ist in beiden Versionen gleich. Wie kann ich aber ein mit dem Amiga geschriebenes Programm auf 5¼-Zoll-Disketten speichern und auf einem PC laufen lassen? Muß ich die Dateien konvertieren? Wie lassen sich die Diskettenformate anpassen?

GÜNTER RABE

MS-DOS-Emulator

Wie komfortabel ist der MS-DOS-Emulator für den Amiga? Welche MS-DOS-Programme können mit einem 5¼-Zoll-Laufwerk geladen werden, alle? Welcher Leser hat mit dem Emulator bereits Erfahrungen gesammelt?

HANS TYEDMERS

Was läuft wie auf dem Amiga 2000?

Vor kurzem verkaufte ich meinen Amiga 500 mit einem zusätzlichen, externen, abschaltbaren Laufwerk. Dafür legte ich mir einen professionellen Amiga 2000 mit zwei internen Laufwerken zu. Die Software behielt ich.

Zu meinem Entsetzen stellte ich fest, daß durch die Speichererweiterung viele Programme auf dem A2000 nicht liefen. Beispielsweise »World Games«, »Winter Games«, »Archon« oder »Two on Two«. Auch »NoFastMem« half nicht. Ferner stellte ich fest, daß einige Programme ein zweites Laufwerk nicht vertragen. Beim A500 war dies kein Problem, aber jetzt — was kann ich machen, um meine Software zum Laufen zu bringen?

RONALD SCHLICHT

Eine Anleitung, um mit Hilfe eines Schalters die Speichererweiterung beim A2000A abzuschalten, finden Sie in der Rubrik »Tips und Tricks«. Für die weiteren Fragen hat vielleicht schon ein Bastler eine Lösung. (ub)

Der Drucker will noch nicht

Wie kann ich den Drucker Riteman F+ am Amiga betreiben? Welche DIP-Schalterstellung ist erforderlich? Welchen Treiber muß ich in den Preferences einstellen?

UWE EHRLICH

ANTWORTEN SIE

Haben Sie schon eine Lösung zu einer der Fragen der Leser. Schicken Sie Ihre Antworten an das Leserforum, damit alle Leser von Ihrem Wissen profitieren. Umfangreiche Vorschläge werden wir eventuell auch in der Rubrik Tips und Tricks veröffentlichen.

IHRE MEINUNG

Schreiben Sie uns Ihre Meinung. Welches Problem brennt Ihnen unter den Fingernägeln?
— Schutz vor Viren,
— Software, die nicht läuft oder nur mit ganz bestimmten Gerätekonfigurationen funktioniert.
— Was halten Sie von der Praxis, Computerprogramme zu indizieren?
— Was gefällt Ihnen an der AMIGA (oder auch nicht)?

Vokabeln gelernt

Mir kam das Programm »Vokabeltrainer« aus der AMIGA 8/9 sehr entgegen, denn ich habe gerade mit dem Englisch begonnen. Jetzt suche ich jemand, der weiß, wie ich die eingegebenen Vokabeln auf einem Drucker ausgeben kann. Wie muß ich das Listing verändern?

STEFANO RANCAN

Druckeranschluß

Ich besitze einen Amiga 500 und habe mir vor kurzem einen NEC P6-Drucker gekauft. Als ich ihn angeschlossen habe, ist mir eingefallen, daß ich einen Druckertreiber brauche. Bei meinem Händler wurde mir mitgeteilt, daß ich den auf der Extra-Diskette befindlichen MPS-2xxx-Treiber verwenden kann. Bei mir funktioniert dies nicht. Von Basic und vom Notepad kann ich keine Daten senden. Können Sie mir weiterhelfen?

MARCO LEBSKI

Der Druckertreiber, den Sie verwenden, ist richtig. Der Fehler muß irgendwo anders liegen. Überprüfen Sie zunächst einmal den Anschluß des Druckers. Haben Sie den Druckertreiber in den Preferences auch korrekt eingestellt? Schauen Sie sich das Menü noch einmal genau an. Anhand des Benutzerhandbuchs können Sie alle Einstellungen vornehmen.

Wenn Sie drucken möchten, benötigt der Amiga neben dem Druckertreiber zusätzliche Informationen. Er braucht drei Dinge, wenn Sie beispielsweise einen Drucker am Parallel-Port angeschlossen haben: die »system-configuration«, das »printer-device« und das »parallel-device«. Auf diese drei Dateien muß der Amiga zugreifen können. Sind alle auf Ihrer Startdiskette? Überprüfen Sie auch, ob die Dateien in den richtigen Ordnern stehen. Der Treiber steht im Verzeichnis »devs/printers«, die drei anderen in »devs«. (ub)

Druckertreiber

In Ausgabe 12/87 auf Seite 81 in der Rubrik Tips und Tricks ist ein Listing abgedruckt, das den Epson-Druckertreiber so verbessert, daß für ein Hardcopy 1920 Punkte pro Zeile verwendet werden. Leider funktioniert das Programm nicht mit meinem Epson-Treiber, dessen Länge 5396 und nicht 5364 Byte beträgt. Liegt es an der Workbench-Version?

ANDREAS DORN

Es sind zur Zeit mehrere Treiber für Epson-Drucker im Umlauf. Das Basic-Programm aus der AMIGA 11 verändert nur einen der älteren Treiber, der wie im Listing angegeben, eine Länge von 5364 Byte besitzt. In den Tips und Tricks der nächsten Ausgabe finden Sie ein Listing, um auch den neuen Treiber zu verändern. (ub)

Zu kleines Basic-Fenster

Nach meiner Information sollte das Ausgabefenster eine Höhe von 200 Punkten haben. Mein Ausgabe-Fenster hat aber eine Höhe von nur 187 Punkten und läßt sich nicht vergrößern.

Ausgabe 11
HANS FELBER

Die bisherige Version des Amiga-Basic verbietet es tatsächlich, auf der Workbench ein Fenster in voller PAL-Höhe zu öffnen. Der Interpreter prüft hier offensichtlich noch auf eine starre NTSC-Maximalhöhe. Als Ausweg bleibt, für dieses Fenster einen eigenen Screen zu öffnen. Das kostet nur wenig Speicherplatz.

Daß Sie bei einer Brutto-Höhe von 200 Punkten nur 187 benutzen können, liegt einmal an der Titelleiste und zum anderen an dem unbeschreibbaren unteren und rechten Rand. Er tritt auf, sobald ein Größen-Symbol vorhanden ist. Letzteres müssen Sie sich in seinen Ausmaßen nach links und oben verlängert vorstellen.

DR. PETER KITTEL
Commodore

Workbench Message

Ich möchte wissen, wie man eine Message von der Workbench, zum Beispiel wenn ich ein Icon anklicke, in C auslesen kann. Die »WB-Startup-Structure« der Workbench enthält zwar viel Interessantes, aber ich weiß nicht genau, wie ich die Daten mit »GetMsg()« auslesen kann. Ich muß wissen, welches Laufwerk zu einem Icon gehört. Kann ich diese Information aus der Message gewinnen? Übrigens habe ich in vielen Büchern nachgeschaut, aber nichts gefunden.

REMO EICHENBERGER

Sie können auf die gewünschten Daten ohne weiteres zugreifen. Sie sind auch bereits auf dem richtigen Weg. Allerdings benötigen Sie »GetMsg()« nicht, wenn Sie in C programmieren und ein Programm von der Workbench starten, um die Startup-Message zu lesen, die das Workbench-Programm an Ihr Programm sendet. Jedes Programm wird vom Linker mit dem Startup-Code versorgt. In diesem Teil wird automatisch die »WBenchMsg« gelesen und ein Zeiger auf diese eingerichtet. Schauen Sie sich den Quell-Text des Startup-Codes auf Ihrer Compiler-Diskette einmal an. Dort stehen beispielsweise folgende Zeilen:

```
WaitPort(&pp->
pr_MsgPort);
WBenchMsg = _GetMsg(&pp->
pr_MsgPort);
if (WBenchMsg->
sm_ArgList)
_CurrentDir(WBenchMsg->
sm_ArgList->wa_Lock);
```

In der ersten Zeile wird auf eine Nachricht vom Workbench-Programm gewartet. »WBenchMsg« stellt nun den Zeiger auf diese Botschaft dar — die »WBStartup-Structure«.

»sm__ArgList« ist ein Zeiger auf eine weitere Struktur:

```
struct WBArg {
 BPTR wa_Lock;
 BYTE *wa_Name;
};
```

»Lock« ist — im wahrsten Sinne des Wortes — der Schlüssel zum vollständigen Dateiverzeichnis des Icons, das Sie angeklickt haben. Im Startup-Code wird das Verzeichnis des Icons übrigens automatisch mit »CurrentDir« zum aktuellen Directory des laufenden Tasks erklärt. (ub)

Return in Basic?

Unter mir noch nicht ganz bekannten Umständen gerät der Amiga beim Arbeiten mit Amiga-Basic in einen Zustand, bei dem er die Return-Taste nicht mehr als Eingabebestätigung erkennt, sondern nur noch als Sonderzeichen auslegt. Am Bildschirm erscheint dann ein offenes Viereck. Selbst ein Warmstart bringt hier keine Abhilfe. Woran liegt das?

RAINER BRUHN
Ausgabe 11/87

Bei mir trat dieses Problem — wenn auch sehr selten — ebenfalls auf. Durch die Eingabe von »Poke 1,1« im Direktmodus läßt sich dieser Effekt sogar absichtlich herbeiführen. Geben Sie »POKE 1,0« ein, dient die Return-Taste wieder dem Zeilenvorschub.

CARSTEN EISEN

Astronomieprogramm gesucht

Wer kennt ein Astronomieprogramm für den Amiga 500, ähnlich wie »Sky Travel« für den C 64. Ich denke, daß sich mehrere Amiga-Besitzer für Astronomie interessieren. Das Programm sollte in der Lage sein, ein Planetarium zu simulieren.

MICHAEL GOTTWALD

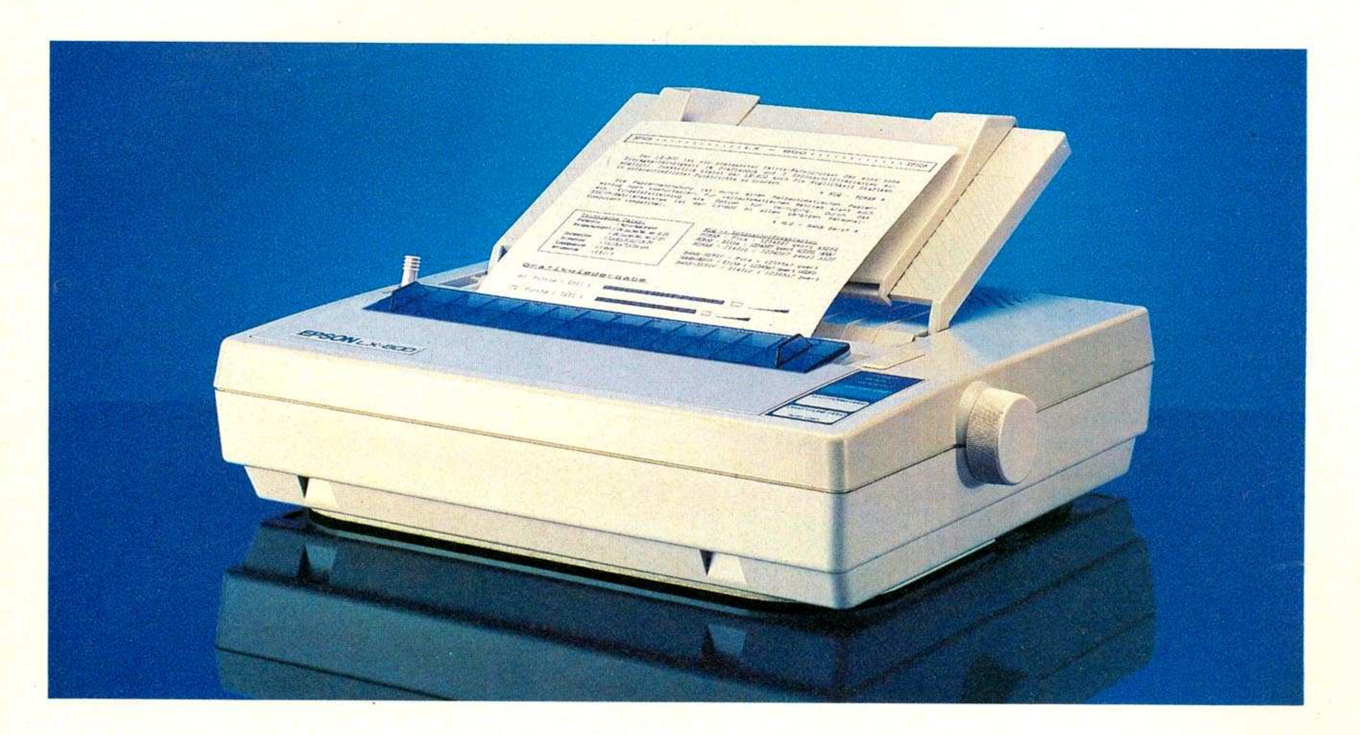
In der AMIGA 12 finden Sie ein Astronomie-Programm zum Abtippen. Ein weiteres Programm dieser Gattung ist »Galileo«. Andere Astronomie-Programme sind uns nicht bekannt. Eventuell besitzt ein Astronomie-Fan mehr Informationen und kann Ihnen schreiben, welches Programm Ihren Wünschen am weitesten entgegenkommt. (ub)

IHRE ANWENDUNG

Auch das ist für die Leser von Interesse. Schreiben Sie ans Leserforum, wie Sie Ihren Amiga nutzen: Privat oder kommerziell? Wenn Sie meinen, eine besondere Anwendung gefunden zu haben — einschicken. Wir werden aus allen Einsendungen die interessantesten herauswählen und veröffentlichen. Bei allen Beiträgen zum Leserforum behalten wir uns allerdings vor, diese aus Platzgründen gekürzt wiederzugeben.

EPSON. Der Unterschied.

M. L. & S.



Es war gar nicht so leicht, EPSON Qualität zu so günstigem Preis im neuen LX-800 zu verwirklichen.

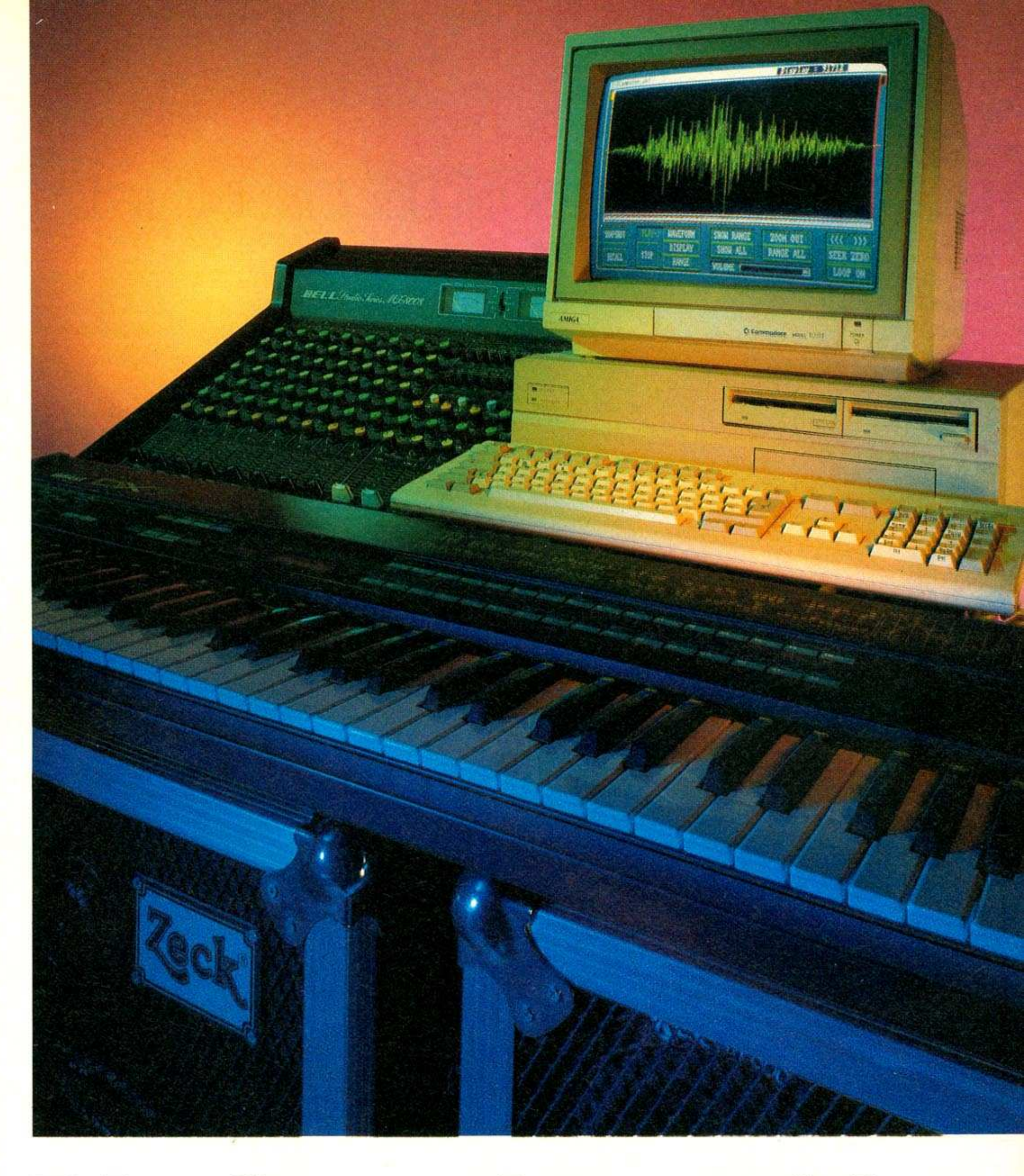
Jetzt kann sich wirklich jeder EPSON Qualität leisten. Dafür sorgt unser neuer LX-Drucker mit seinem günstigen Preis. Bei erstaunlich niedrigem Geräuschpegel erreicht er ein Top-Tempo von 180 Z./Sek. Und auch in Schönschrift (NLQ) bewältigt er einen für seine Klasse extrem hohen Textdurchsatz. Den Unterschied macht aber auch die Ausstattung deutlich: Zwei Schönschriften, vielfältige Schriftvarianten, volle Grafikfähigkeit, übersichtliches Bedienerfeld, Endlospapier- und Einzelblattverarbeitung. Und auf Wunsch ein vollautomatischer Einzelblatt-Einzug. Über die parallele Schnittstelle läßt sich der LX-800 an alle gängigen PCs anschließen. Für Commodore-Anwender gibt es den LX-800 mit zusätzlich eingebauter C64/128 Schnittstelle. Er ist also ein ideales Aufsteiger-Modell. Und der Preis macht das Einsteigen leicht. Fragen Sie Ihren EPSON Fachhändler.





Technologie, die Zeichen setzt.

EPSON Deutschland GmbH · Zülpicher Straße 6 · 4000 Düsseldorf 11 · Telefon 0211/5603-0



Die Soundmaschine

Der Amiga ist durch seine Hardware-Ausstattung prädestiniert für Anwendungen wie Soundsampling, computerunterstütztes Komponieren und Ansteuerung von MIDI-Hardware.

Der Soundchip »Paula« im Amiga unterscheidet sich wesentlich von dem des C 64 oder Atari ST. Es gibt nicht mehr Dreieck, Rechteck, Sägezahn und Rauschen als Wellenformen, da der Klang ganz anders zustande kommt. Das Zauberwort heißt Soundsampling und ist eine erst in letzter Zeit populär und preisgünstig gewordene Technologie. Sounds werden nicht mehr synthetisch erzeugt, sondern in der Natur vorkommende Klänge werden mit einem Mikrofon in elektrische Schwingungen und mit dem Soundsampler in digitale Daten umgewandelt. Schickt man diese digitalen Informationen in eine entsprechende Schaltung, so wird der ursprüngliche Klang wieder hörbar. Zuerst muß untersucht werden, wie sich ein Klang aufbaut. Man hat herausgefunden, daß sich jeder Klang aus unendlich vielen Sinusschwingungen verschiedener Frequenz und Amplitude zusammensetzt. Für den Menschen ist jedoch nur ein Bereich von ungefähr 16 bis 20000 Hertz hörbar, das heißt nur dieser Frequenzbereich muß berücksichtigt werden. In Bild 1 ist der Vorgang des Digitalisierens und der Wiedergabe der so gewonnenen Toninformationen in einer groben Übersicht dargestellt. Nach der Aufnahme des Klangs durch das Mikrophon (im Bild links) folgt im Sampler der erste Schritt zur Digitalisierung:

— Der »Tiefpaßfilter« (Filter, der nur tiefe Frequenzen passieren läßt) sorgt dafür, daß nur der Frequenzbereich in den Sampler gelangt, der später auch wiedergegeben werden soll (Bild 2). Ist kein Filter vorhanden, kann es zu Verzerrungen kommen.

— Nun wird die Wellenform (Bild 3) mit einer bestimmten Rate (der »Sampling-Rate«) abgetastet. Innerhalb eines Zyklus der Abtastung wird ein Amplitudenwert der Wellenform am Ausgang dieses Moduls konstant gehalten. Daher auch der Name »Sample & Hold« (abtasten und halten).

— Dies ist nötig, da das folgende Modul, der »Analog-Digital-Wandler«, einen Taktzyklus lang diesen Wert benötigt, um ihn in einen digitalen Wert, also eine Zahl, umzuwandeln. Die erzeugten Werte hängen von der Auflösung des Systems ab. Im Amiga beträgt diese 8 Bit, damit kann die Zahl 256 verschiedene Werte annehmen.

— Die Zahl wird vom Prozessor aufgenommen und in den Speicher geschrieben. Damit ist der Samplevorgang beendet. Der Klang liegt nun im Speicher in codierten Amplitudenwerten vor.

Der Amiga ist nur mit zusätzlicher Hardware in der Lage, Töne zu digitalisieren, da der Eingangstiefpaßfilter, das Sample & Hold und der Analog-Digital-Wandler nicht vorhanden sind. Anders verhält es sich beim Abspielen von Sounds. Hier ist alles schon eingebaut, was benötigt wird. Allerdings muß beim Amiga nicht jeder Wert einzeln an den Digital-Analog-Wandler (Gegenstück zum Analog-Digital-Wandler) übergeben werden. Das erledigt der Soundchip Paula von selbst. Mit seinem DMA-Zugriff (Direct Memory Access = Direkter Speicherzugriff) muß der Prozessor ihm nur mitteilen, welchen Speicherbereich er wie schnell (Sampling-Rate) ausgeben soll.

Der Superchip Paula

Jedoch muß hinter den Digital-Analog-Wandler wieder ein Tiefpaßfilter geschaltet sein, da ansonsten das sogenannte »Quantisierungsrauschen« enorm stört. Dies sind Nebengeräusche, die entstehen, wenn ein Sampling-Wert vom vorhergehenden sehr verschieden ist. Dies kann durch den Tiefpaßfilter ausgeglichen werden. Leider ist dieser im Amiga fest eingestellt und (zumindest bei den deutschen Amiga-Modellen) nicht abschaltbar, was den Sound unter Umständen dumpf erscheinen läßt, obwohl mit hoher Sampling-Rate gearbeitet wurde. Paula hat vier solcher Digital-Analog-Wandler, deren Ausgangskanäle paarweise auf zwei Anschlüsse zusammengefaßt sind. Deshalb ist der Amiga in der Lage, echten Stereo-Sound zu produzieren. Jeder Kanal kann durch ein Register (also eine Speicherstelle in Paula) in der Lautstärke geregelt werden. Jeder dieser Kanäle ist sowohl in der Lautstärke als auch in der Abspielfrequenz voll modulierbar. Mit der Amplitudenmodulation ist bei einer langen, kontinuierlichen Modulationswellenform eine Hüllkurvensteuerung wie bei

konventionellen Synthesizern möglich. Mit der Frequenzmodulation sind bei geschickter Auswahl von Wellenformen und -frequenzen sehr außergewöhnliche Klänge möglich.

Alle diese Eigenschaften lassen sich komfortabel mit dem Programm »DeLuxe Sound« (Test AMIGA 12/87) einstellen. Dies ist eine Samplersoftware inklusive der benötigten Zusatzhardware, bei der die oben erwähnten AM- und FM-Wellenformen mit Hilfe der Maus grafisch eingegeben werden können. Neben der Möglichkeit, die Abspielgeschwindigkeit während der Wiedergabe zu verändern, kann der Klang auch mit einem Echo-Effekt versehen werden.

Amiga-Sampler

Alle diese Modulationen können auch in Realtime, das heißt direkt beim Digitalisieren, verwendet werden. Dabei ergibt sich die Möglichkeit, den Amiga als Super-Effektgerät zu verwenden, da das digitale Signal sofort von den angewählten »Effekten« bearbeitet und ausgegeben wird. Man kann ein Sample auch direkt auf Diskette speichern und dieses dann wieder direkt von Disk abspielen. Soll ein gleichförmig klingender Sound erstellt werden, zum Beispiel eine Orgel, so muß nicht die gesamte Tondauer aufgenommen werden, sondern nur der Anfang und ein kleines Stück des gleichförmigen Klanges. Mit zwei Schiebereglern kann man bei DeLuxe Sound (und auch bei jedem anderen hochwertigen Sampler) nun einen Teil der Aufnahme bestimmen, der nach Abspielen des Anfangs ständig wiederholt wird. Durch dieses Verfahren, auch Looping genannt, spart man sich eine Menge wertvollen Speicherplatz. Allerdings ist es wichtig, daß am Anfang und am Ende des »Loops« ein Nulldurchlauf (also der Wert 0) vorhanden ist, da es sonst zu störenden Verzerrungen kommt. Hierzu wäre eine grafische Ausgabe des Samples vonnöten, die jedoch bei DeLuxe Sound nicht vorhanden ist.

Ein Problem der Sampling-Technik ist, daß ein natürlicher Klang, etwa ein Klavier, nicht in jeder Tonlage gleich klingt. Da ein herkömmlicher Digitalisierer aber verschiedene Tonhöhen dadurch erzeugt, indem er den Klang einfach verschieden schnell ausgibt, wird dies nicht berücksichtigt. Dieses Problem tritt besonders deutlich bei der menschlichen Stimme auf. Aus einer digitalisierten normalen Stimme wird in höheren Oktavlagen immer eine Art Micky-Maus-Stimme. Um dies zu verhindern, können manche Sampler einen Klang in verschiedenen Tonlagen aufnehmen, und verwenden bei der Wiedergabe immer die Aufnahme, die der gewünschten Tonhöhe am nächsten kommt. Diesen Trick nennt man »Multisampling«. Der Nachteil bei diesem Verfahren ist ein erhöhter Speicherplatzverbrauch, da für einen einzelnen Klang mehrere Samples benötigt werden. Professionelle Tondigitalisierer in Preisklassen um die 40000 Mark verwenden Multisampling auch noch für verschiedene Lautstärken, da ein leise angeschlagenes Klavier wiederum anders klingt als ein lauter gespieltes. So kommt es, daß selbst Konzertpianisten kaum noch den Unterschied zwischen einem echten und einem digital aufgenommenen Flügel heraushören können. Eine solche naturgetreue Wiedergabe scheitert beim Amiga vor allem an der zu niedrigen Auflösung der D-A-Wandler, die nur 8 Bit beträgt.

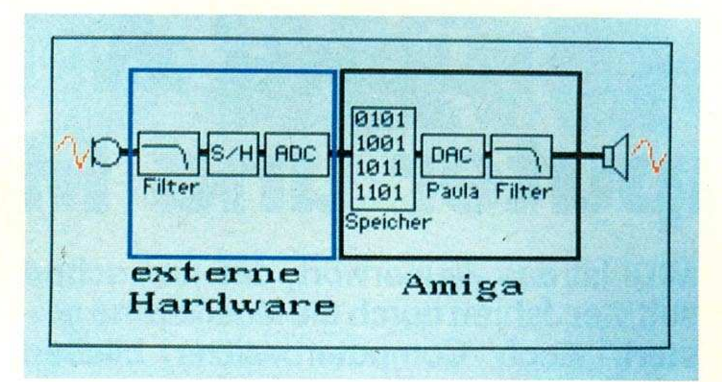


Bild 1. Die an der Sound-Digitalisierung beteiligten Einheiten im Sampler und Amiga in der Übersicht

Bild 2. So stellt sich ein Klang im Frequenzgang dar

Um alle Amiga-Dateien möglichst für jedes Programm verwendbar zu machen, haben Commodore und Electronic Arts schon 1985 den »IFF-Standard« entwickelt. So gibt es auch einen Standard für digitale Tondaten, der »8 Bit Sampled Voice« (8SVX) heißt. Mit ihm ist es möglich, sowohl Effekt-Sounds als auch Instrumente mit oder ohne Looping zu speichern. Es sind Multisamples mit einem Sample pro Oktave möglich. Außerdem kann eine Lautstärke und eine Amplitudenhüllkurve angegeben werden. Fast jedes Musikprogramm für den Amiga kann mit dem 8SVX-Standard arbeiten. So auch das »Deluxe-Music Construction Set«. Seine große Stärke ist die perfekte Notendarstellung in einem bis zu vierfachen Notensystem. Es sind alle erdenklichen Noten- und Pausenwerte, Ton- und Taktarten, Notenschlüssel, Spielweisen, Lautstärken möglich. Jedem Notensystem wird ein IFF-Sound oder ein MIDI-Kanal zugeordnet (MIDI ist eine genormte serielle Schnittstelle für professionelle Synthesizer und Computer). Bei der Einstellung »MIDI« werden die Noten dann nicht mit dem Amiga-Soundchip gespielt, sondern an einen externen MIDI-Synthesizer geschickt. Die Noten können sogar per MIDI-Tastatur eingespielt werden und müssen nicht mühsam Note für Note mit der Maus plaziert werden. IFF-Sounds lassen sich mit Deluxe-Music allerdings nicht programmieren; es stehen nur bereits vorhandene Sounds zur Auswahl. Auf einem ganz anderen Konzept basiert Sonix von Aegis. Neben normalen »Sample-Sounds kann ein kon-

Fortsetzung auf Seite 118

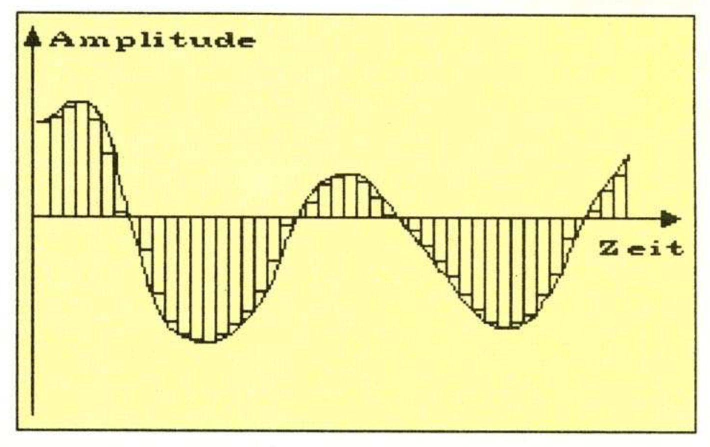


Bild 3. Die Qualität der Digitalisierung hängt im wesentlichen von der Sampling-Rate ab (Abtastauflösung)

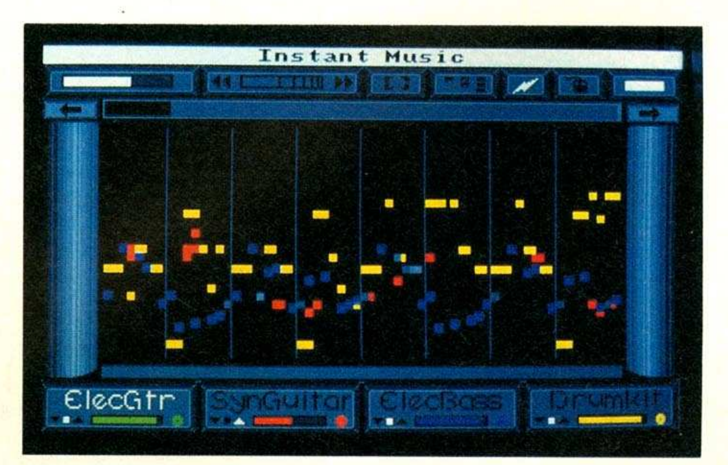


Bild 4. »Instant Music« fällt durch die Notendarstellung und das hilfreiche Mouse-Jam aus dem Rahmen

MIDI — Brücke zwischen Computer und Musik

MIDI ist das Zauberwort, das nun schon seit vier Jahren durch die Musikszene geistert. Auch Computerbesitzer blieben von der MIDI-Welle nicht ausgenommen, da Interfaces für die verschiedensten Computer entwickelt wurden. Was MIDI ist und was dieser Standard alles kann, erfahren Sie in diesem Bericht.

Bei den ersten monophonen (einstimmigen) Synthesizern war alles noch relativ einfach. Mit Hilfe einer Steuerspannung wurde die Tonhöhe bestimmt. Dabei hatten sich die Pioniere der Synthesizertechnik auf das »1 Volt pro Oktave-System« geeinigt. Dieser Standard besagt, daß aus einer Spannungsverdoppelung auch eine Verdoppelung der Tonfrequenz resultiert. Doch als die ersten polyphonen (mehrstimmigen) Synthesizer auf den Markt kamen, vollzog sich ein Wandel. Diese Geräte waren meist viel kompakter als die einstimmigen Modulsysteme und hatten eine wesentlich aufwendigere Tastaturabfrage. Mit Einzug der neuen Technik verloren die Firmen den gemeinsamen Standard aus den Augen und entwickelten unterschiedliche Systeme.

Der Vater von MIDI

Diese sehr oft nicht mehr kompatiblen Systeme führten zu einer wachsenden Unzufriedenheit bei den Musikern. Auf der Bühne, aber auch im Studio, waren sie darauf angewiesen, daß sich unterschiedliche Geräte koppeln ließen. Einer, der sich um dieses Problem maßgeblich Gedanken machte, war Dave Smith. Der Chef und Gründer der amerikanischen Synthesizerfirma Sequential Circuits legte bereits 1981 das Konzept einer einheitlichen Schnittstelle für Keyboards vor. Er nahm Kontakt zu allen führenden Synthesizerherstellern auf, zwecks Verabschiedung eines einheitlichen Standards. Enttäuscht mußte er jedoch feststellen, daß seine amerikanischen Kollegen weit weniger für die Idee zu gewinnen waren, als japanische Hersteller. Die wesentliche Entwicklungsarbeit des Konzepts verteilte sich deshalb auf Sequential Circuits und die japanische High-Tech-Firma Roland. Anfang 1983 wurden auf der Winter-Namm-Show (USA) die ersten mit MIDI ausgestatteten Synthesizer vorgestellt. Erst im August 1983 wurde eine offizielle MIDI-Spezifikation (V1.0) herausgegeben. Diese detaillierte Beschreibung, die bis heute einige Male erweitert worden ist, stellt eine Art Gesetz dar, an das sich alle Hersteller von MIDI-Equipment halten sollten.

MIDI-Hardware

Rein äußerlich könnte man die MIDI-Schnittstelle leicht mit einem Audioausgang verwechseln, da eine 5polige Diodenbuchse wie bei älteren HiFi-Anlagen verwendet wird.

Bild 1. Mehrere Synthesizer werden in einer Kette verbunden für den einfachen MIDI-Omni-Mode

Doch MIDI hat mit einem Analogausgang nicht das geringste gemeinsam, da es sich um eine digitale, serielle Schnittstelle handelt. Die Interfaceschaltung erzeugt eine 5mA-Stromschleife, die in den Empfängerteilen (MIDI-In) über Optokoppler läuft. Das MIDI-Equipment ist also elektrisch überhaupt nicht miteinander verbunden. Dadurch sind die Geräte relativ gut vor Fehlanschlüssen und Brummspannungen geschützt. Die Übertragungsrate dieses Standards beträgt 31 250 Bit/s, wobei die Daten immer in Päckchen zu je 10 Bit übertragen werden:
1 Startbit/8 Datenbit/1 Stoppbit

Diese zusammengefaßten Informationen können durch 16 adressierbare Kanäle geschickt werden. Dies hat den Vorteil, daß in einem größeren MIDI-Verbund (etwa mehrere Synthesizer) Daten gezielt an ein bestimmtes Gerät adressiert werden können.

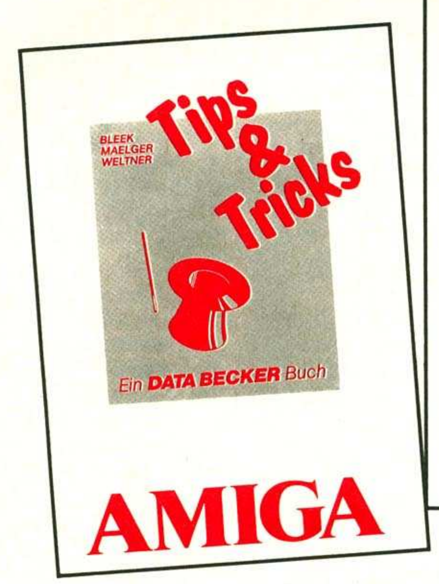
Dazu muß das betroffene Keyboard natürlich auf dem entsprechenden Kanal auf Empfang geschaltet sein. Mit Hilfe dieser 16 Kanäle lassen sich prinzipiell 16 verschiedene MIDI-Instrumente voneinander unabhängig steuern. Der MIDI-Standard differenziert jedoch nicht nur verschiedene Kanäle, sondern auch unterschiedliche Informationsarten. Die Informationen werden eingeteilt in Systemdaten und Kanaldaten. Der Kanaldatentransfer wird durch bestimmte Betriebsarten gesteuert. Im einzelnen verwendet der MIDI-Standard die Modi OMNI, POLY, MONO und MULTI:

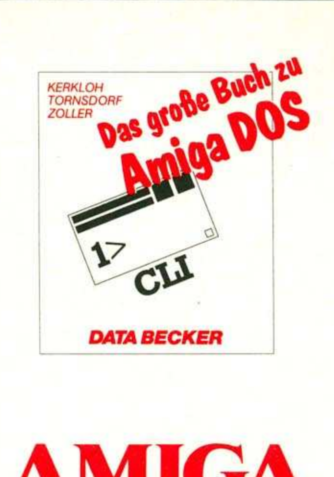
Omni-Mode: In dieser Betriebsart wird nicht zwischen einzelnen Kanälen unterschieden. Jedes Instrument sendet im Normalfall auf Kanal 1 und empfängt auf Kanal 1 bis 16. In der Praxis bedeutet dies, daß alle am Synthesizer A angespielten Noten mit der gleichen Programmnummer auch auf dem Synthesizer B erklingen (Bild 1). Das heißt nicht, daß beide Keyboards auch den gleichen Sound haben, da MIDI nur die Information des angewählten Programms überträgt, nicht aber den Klang selbst.

Poly-Mode: Dieser Modus ist gegenüber dem Omni-Mode schon weit fortschrittlicher, da sich bei jedem Gerät festlegen läßt, auf welchem Kanal die Daten gesendet und empfangen werden können. Damit ist die unabhängige Steuerung mehrerer Instrumente kein Problem mehr (Bild 2). Auf jedem Instrument kann eine andere MIDI-Kanalnummer eingestellt werden. Beispielsweise wird dem Synthesizer A der Kanal 1 zugeordnet, dem Synthesizer B der 2. Kanal und einem Sampler der Kanal 4. Theoretisch lassen sich allen 16 Kanälen verschiedene MIDI-fähige Instrumente zuordnen, die alle einzeln adressierbar sind.

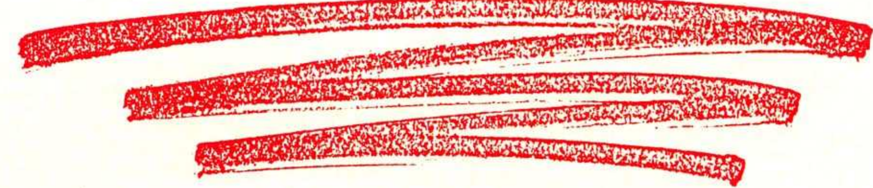
Mono-Mode: Diese Betriebsart unterscheidet sich wesentlich von den bereits genannten und war ursprünglich teuren Instrumenten vorbehalten. Beim Poly-Mode ist es möglich, diverse Keyboards mit unterschiedlichen Sounds anzusteuern. Einen polyphonen Synthesizer gleichzeitig mit mehreren verschiedenen Klängen spielen zu lassen, ist aber nicht realisierbar. Der Mono-Mode öffnet in dieser Hinsicht völlig neue Perspektiven. In dieser Betriebsart kann jeder Stimme des Synthesizers ein eigener MIDI-Kanal zugeordnet werden. Bei einem sechsstimmigen Keyboard könnten die Kanäle 1 bis 6 vergeben werden. Der erste Kanal wäre in diesem Fall der Basiskanal, zu dem die restlichen

Top aktuell:









Amiga Tips & Tricks – jetzt in einer völlig überarbeiteten Neuauflage. Hier verraten Ihnen echte Profis, mit welchen Tricks sie mehr aus dem Amiga holen: Hilfen zur Gestaltung eigener Programme, Tips & Tricks zum AmigaBASIC, Maschinenprogramme in AmigaBASIC, Einsatz von DOS-Routinen, optimierende Hilfsprogramme für AmigaBASIC-Programme, Tips zur Arbeit mit der Workbench, Aufbau der Icons, neue Ein-/Ausgaberoutine. Mit vielen Anregungen, aber auch fertigen Lösungen. Greifen Sie in die Trickkiste, und schon werden Dinge wahr, die Sie nicht für möglich hielten. Ein Buch, das voller Überraschungen steckt.
Amiga Tips & Tricks – die riesige Fundgrube für jeden Amiga-Besitzer.

Amiga Tips & Tricks
Hardcover, 473 Seiten
DM 49,–

Der Amiga macht es einem so leicht wie möglich. Nahezu alles läßt sich problemlos über die Workbench bearbeiten. Wenn Sie jedoch den Mut haben, die komfortable Oberfläche zu verlassen, werden Sie schon sehr bald belohnt – mit einigen Dingen, die Sie dem Amiga bisher nicht zugetraut hätten. Das große Buch zu AmigaDOS hilft Ihnen dabei. Neben einem ausführlichen Einsteigerteil erfahren Sie alles, was Sie bei Ihrer praktischen Arbeit mit dem AmigaDOS wissen sollten: Umlenken der Ein- und Ausgabe, sinnvoller Einsatz des Jokers, Arbeiten mit RAM-Disk und CLI, Batch-Dateien, STARTUP-Sequenz, Multitasking mit dem CLI, Aufbau der CLI-Befehle, Programmierung eigener CLI-Befehle, neue CLI-Befehle in BASIC und C. Dazu ein ausführlicher, gut strukturierter Nachschlageteil. Wer also mit dem AmigaDOS arbeiten möchte, sollte dieses Buch immer in greifbarer Nähe haben.

Das große Buch zu AmigaDOS
Hardcover, 320 Seiten
DM 49,–

Das Buch, das zur Amiga-Floppy keine Frage offenläßt. Hier finden Sie Dinge, die Sie im Handbuch vergeblich suchen werden: Floppy-Operationen unter der Workbench und unter AmigaDOS im CLI, relative und sequentielle Dateien, Aufbau der Diskette, Zugriff über Trackdisk-Device, Track lesen und schreiben, Kodier-und Dekodierroutinen des Betriebssystems… Mit vielen nützlichen Programmen wie z. B. ein Superkopierprogramm oder einen Floppyspeeder. Was Sie wissen müssen, finden Sie hier – vom Einsteiger zum Profi.

Amiga Floppy Buch
Hardcover, ca. 350 Seiten
inkl. Diskette, DM 59,–
erscheint ca. 1/88

BESTELL-COUPON
Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
unabhängig von der bestellten Stückzahl
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name
Straße
Ort

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

Kanäle aufsteigend benachbart sein müssen. Es ist also nicht möglich, Kanal 1 mit den Kanälen 3 bis 7 zu kombinieren.

In Verbindung mit Sequenzern oder Computern sind sehr komplexe Arrangements realisierbar. Ein kleiner Nachteil des Mono-Mode ist, daß beispielsweise für einen dreistimmigen Akkord mit demselben Sound drei MIDI-Kanäle belegt werden müssen.

Multi-Mode: Dieser Modus ist noch relativ neu und doch am weitesten fortgeschritten. Er vereint jeweils die Vorteile des Poly- und Mono-Modes (Bild 3). Im Prinzip verhält sich ein Synthesizer, der über den Multi-Mode verfügt, wie ein System mehrerer Keyboards im Poly-Mode. Das heißt: es lassen sich über verschiedene Kanäle die einzelnen Stimmen des Synthesizers mit unterschiedlichen Sounds ansteuern. Dabei können im Gegensatz zum Mono-Mode über einen Kanal auch mehrere Stimmen mit dem gleichen Sound (Akkorde) übertragen werden. Die Kanäle müssen in dieser Betriebsart auch nicht mehr benachbart sein.

Diese vier Betriebsarten sind nicht bei allen Synthesizern zu finden. Standard ist bis heute eigentlich erst der Poly-Mode.

ihre Stimmung zu prüfen und gegebenenfalls zu korrigieren. Ebenso wichtig wie die richtige Stimmung des Equipments ist die korrekte Zeitsynchronisierung rhythmischer Abläufe. Damit das MIDI-Equipment nicht aus dem Takt gerät, dienen die System-Realtime-Informationen. Einem Zusammenwirken verschiedener Drumcomputer, Sequenzer und Synthesizer steht also dank MIDI kein größeres Hindernis im Wege. Inzwischen gibt es sogar über MIDI steuerbare Mischpulte, die diese

System-Informationen

Echtzeitinformationen in computergesteuerte Mischvorgänge umsetzen. Einen Sonderstatus nehmen die System-Exclusive-Daten ein. Moderne Synthesizer werden immer komplexer und damit oft auch sehr schwer bedienbar. Die vielen analogen Regler bei älteren Synthesizern mußten mehr und mehr den Tastern und Displays fortschrittlicher Digitaltechnik weichen. Diese Entwicklung half die Kosten der Instrumente drastisch zu senken, der Bedienungskomfort blieb aber leider oft auf der Strecke. Was lag näher als der Einsatz eines Computers, um den Bedienungskomfort wieder zu heben. Fast zu jedem MIDI-Synthesizer existieren entsprechende Editor-Programme, die um die Gunst der Käufer ringen.

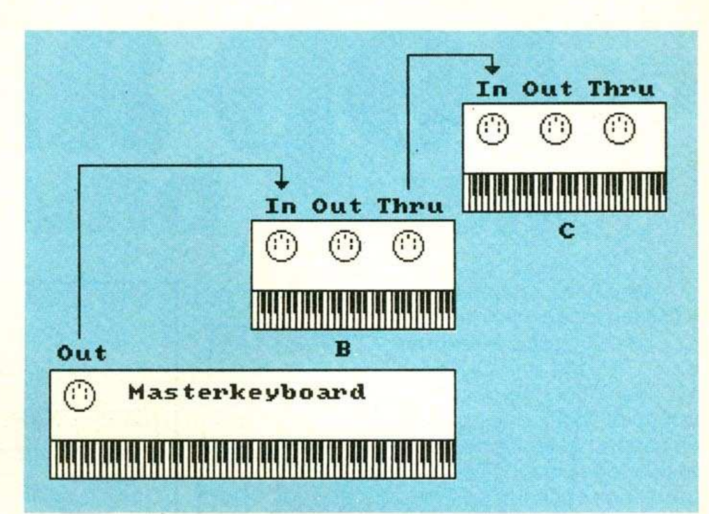


Bild 3. Die typische Konfiguration für einen Betrieb im Multi-Mode mit einem Master-Keyboard

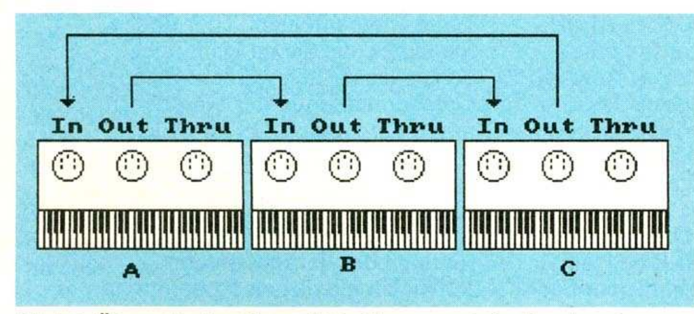


Bild 2. Über die 5poligen Kabel lassen sich die einzelnen MIDI-Geräte auch im Kreis verbinden

Im Gegensatz zu den Kanaldaten (Channel-Voice- und Channel-Mode-Informationen), die meistens nur ausgewählte Instrumente ansprechen, enthalten die Systemdaten Informationen, die alle Geräte im MIDI-Verbund betreffen, gleichgültig in welchem Modus sie sich befinden. Es wird im einzelnen zwischen System-Common-, System-Realtime- und System-Exclusive-Informationen getrennt.

Ein typisches Beispiel für einen System-Common-Befehl ist der MIDI-Tune-Request. Dieser Befehl veranlaßt alle angeschlossenen Synthesizer,

Bei diesen Informationen endet die Kompatibilität des MIDI-Standards. Jeder Hersteller darf die möglichen System-Exclusive-Befehle seiner MIDI-Produkte selbst definieren und implementieren. Dies ist durchaus sinnvoll, da die Geräte verschiedener Hersteller oft völlig unterschiedliche Möglichkeiten aufweisen. Mit Hilfe der System-Exclusive-Informationen können Sounds von einem Synthesizer in den Computer transferiert, in diesem modifiziert und anschließend wieder zum Keyboard zurückgeschickt werden. Bei Bedarf lassen sich die Klänge auch auf Diskette archivieren. Im Prinzip kann über die System-Exclusive-Daten der Computer die Herrschaft über das gesamte Tonerzeugungssystem (Oszillatoren, Filter, Hüllkurven) des Synthesizers übernehmen.

Was läuft eigentlich ab, wenn zwei Synthesizer miteinander gekoppelt werden? Das sendende Keyboard (MIDI-OUT) wird als Master bezeichnet, im Gegensatz zum Empfänger (MIDI-IN), der Slave genannt wird. Da es sich nur um zwei Instrumente handelt, kann durchaus der Omni-Mode benützt werden. Werden am Master-Keyboard eine oder mehrere Tasten gedrückt, so gehen diese Informationen unverzüglich auch zum Empfänger-Synthesizer über. Dieser würde ohne hörbare Verzögerung die gleichen Töne wie das sendende Instrument spielen.

Praxiseinsatz

Da in den meisten Fällen unterschiedliche Geräte miteinander verbunden werden, würde das zweite Gerät natürlich mit einem anderen Sound erklingen als das erste. Diese Übereinanderlagerung verschiedener Klänge ist äußerst reizvoll und gehört mitunter zu den Hauptanwendungsgebieten von MIDI. Bei Musikgruppen werden oft sogar mehrere Synthesizer miteinander gekoppelt, um die Klangfülle zu erhöhen. Da bei diesen Zusammenschlüssen die Tastaturen der Slaves eigentlich überflüssig sind, führt der Trend immer mehr in Richtung tastaturlose Expander. Dies sind vollwertige Synthesizer, die meist im 19-Zoll-Format angeboten werden. Die zweite Änderung spielt sich im Bereich der Tastaturen ab. Immer mehr sogenannte Master-Keyboards mit herausragenden MIDI-Möglichkeiten kommen auf den Markt. Diese intelligenten Tastaturen haben oft gewichtete Tasten (ähnlich einem Klavier) und sind voll auf die Koppelung mit mehreren Expandern und Synthesizern ausgelegt. In zunehmendem Maße werden auch Effektgeräte wie digitale Hallgeräte mit der MIDI-Schnittstelle versehen. Dies hat den Vorteil, daß bei Soundänderungen am Synthesizer gleich die richtigen Effekte am Hallgerät mit angewählt werden.

Die tollsten Möglichkeiten entfaltet ein MIDI-System aber erst in Verbindung mit einem Computer. Die Anwendungen reichen von Editor- und Sequenzerprogrammen bis hin zu Notendruck-Applikationen, die vom Keyboard eingespielte Melodien in klassischer Notenschrift auf dem Drucker ausgeben. Allein diese Anwendungen müßten Mozart und Beethoven vor Neid erblassen lassen. Doch die Anwendungsmöglichkeiten gehen noch weiter. Mit den neuen MIDI-Mischpulten ist auch das letzte Glied in der Übertragungskette automatisierbar. Steuert ein Computer einen MIDI-Verbund, so kann ein Arrangement vollautomatisch mit korrekter Abmischung auf Band gespielt werden.

Das Wichtigste, was uns jedoch weder MIDI noch der Computer abnehmen kann, ist immer noch die Fähigkeit, Musikstücke zu komponieren.

(Bernhard Carli/jk)

Tusch mit Trommelwirbel

AMIGA test

»Dynamic Drums« ist eines der ersten professionellen Schlagzeugprogramme für den Amiga. Die Gestaltungsmöglichkeiten machen das Programm als Ersatz für einen Drumcomputer interessant.

Das »Dynamic Drums«-Paket wird mit zwei Disketten ausgeliefert. Dabei ist eine Diskette fast vollständig mit über 60 verschiedenen Drumsounds gefüllt. Leider ist jedoch das Hauptprogramm mit einem so »ausgeklügelten« Kopierschutz versehen worden, daß manche Amiga 500-Besitzer manchmal Schwierigkeiten haben werden, das Programm überhaupt zu laden.

Als eigentliche »Trommeln« fungieren bei Dynamic Drums die Tasten der Zehnertastatur. Jede dieser Zifferntasten wird mit einem Drumsound belegt; für zusätzliche Effekte werden noch einige weitere Tasten des Zehnerblocks benötigt.

Ist das Programm geladen, wird man zuerst zur Auswahl eines von zehn auf der Drum-Diskette vorhandenen »Drumkits« aufgefordert. Ein Drumkit ist eine Zusammenstellung von zehn verschiedenen Trommelsounds, die auf Druck einer der Tasten des Zehnerblocks gespielt werden. Die Belegung der Zehnertastatur wird ständig auf dem Bildschirm angezeigt (Bild 1 links).

Zehn Trommeln im Speicher

Um den Platzverbrauch eines Drumkits auf der Diskette möglichst gering zu halten, werden nur die Namen der zehn verwendeten Drums sowie einige Steuerdaten gespeichert. Das hat allerdings den Nachteil, daß die Drumsounds beim Laden eines Drumkits einzeln in den Speicher geladen werden müssen, was insgesamt etwa 40 Sekunden dauert. Wechselt man also häufig zwischen den Drumkits hin und her, empfiehlt sich die Verwendung des Floppybeschleunigers FACC.

Ist ein Drumkit geladen, kann bereits auf der Zehnertastatur kräftig getrommelt werden. Die ENTER-Taste des Zehnerblocks hat dabei eine besondere Bedeutung: Wird sie zusammen mit einer der Ziffern gedrückt, so ertönt der Trommelschlag der jeweiligen Ziffer in einer anderen Tonhöhe (und eventuell Lautstärke). Auf diese Weise lassen sich »akzentuierte« Schläge spielen. Wenn die »—«-Taste gleichzeitig mit einer Ziffer gedrückt wird, wechselt das Programm bei jedem Trommelschlag zwischen der akzentuierten und der normalen Tonhöhe. Die über 60 Drums, die mit dem Programm auf einer eigenen Diskette mitgeliefert werden, sind sicher für die meisten Anwendungen vollkommen ausreichend. Das Spektrum reicht hier von 19 verschiedenen Baß-Drums über verschiedene High-Hats, Toms und Snares bis hin zu ausgefallenen Effekten wie dem Zersplittern eines Glases. Obwohl die vorhandenen Drumkits von Dynamic Drums bereits einen großen Bereich abdecken, kann man jederzeit auch eine einzelne Trommel laden und damit eine bereits vorhandene auf der Zehnertastatur ersetzen. Sollten die mitgelieferten Sounds nicht ausreichen, kann das Programm alle Dateien mit digitalisierten Sounds einlesen und abspielen. Nicht alle Programme im Bereich der Sound-Digitalisierung erstellen Files (Dateien) nach dem IFF-Standard. Daher sollte man beim Laden von Fremddateien vorsichtig sein; das Programm überprüft nicht, ob eine Datei auch tatsächlich digitalisierte Töne enthält. Zu lange Dateien können zum Absturz des Computers führen.

Das Einlesen von Sounds von anderen Disketten wird dadurch unnötig erschwert, daß Name und Unterverzeichnis der Diskette teilweise per Hand eingetippt werden müssen. Es ist deshalb zu empfehlen, sich — wie im Handbuch beschrieben — eine Sounddiskette anzulegen und alle benötigten Sounds dorthin zu kopieren. Um überhaupt eigene Drumkits speichern zu können, ist dieser Schritt sogar unbedingt notwendig.

Leider wird das sehr umfangreiche Inhaltsverzeichnis der Drumsounds nicht vom Programm zwischengespeichert, was bei jedem Laden einer Trommel eine längere Wartezeit mit sich bringt.

Natürlich liegt die Hauptaufgabe von »Dynamic Drums« nicht beim einfachen Herumtrommeln auf der Zehnertastatur. Für einen Musiker ist es viel wichtiger, daß etwa ein bestimmter Grundrhythmus »aufgenommen«, verändert und wieder abgespielt werden kann. Deshalb wurde in das Programm ein Sequenzer (System zum Aufnehmen und Abspielen von Tönen) eingebaut, der die rechte Hälfte des Bildschirms (Bild 1) einnimmt. Die kleinsten Einheiten, die der Sequenzer abspielen kann, sind die sogenannten »Patterns«. Ein Pattern kann maximal 64 Zweiunddreißigstelnoten aufnehmen, was für einen Grundschlag, der sich wiederholen soll, sicherlich ausreicht.

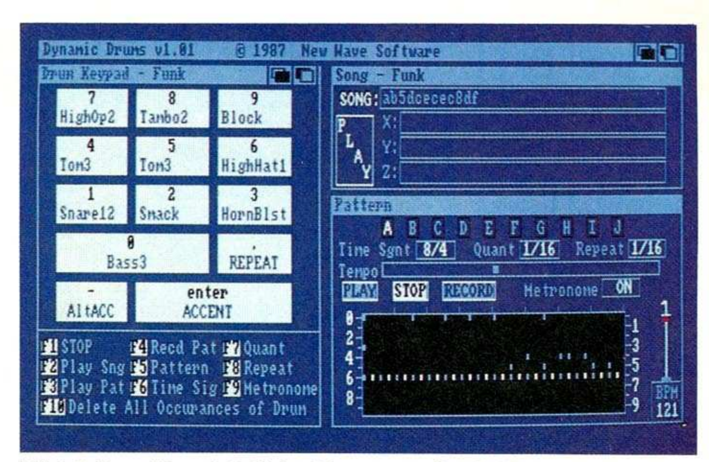


Bild 1. Das übersichtliche Hauptmenü mit Metronom

In dem bereits angesprochenen Fenster werden alle Trommelschläge grafisch als Rechtecke aufgezeichnet, wobei jeder der zehn Trommeln eine Zeile und jedem der 64 möglichen Schläge eine Spalte zugeordnet wird. Normal gespielte Schläge sind blau und akzentuiert gespielte weiß gefärbt. Das Programm erlaubt die Bearbeitung von 3/4, 4/4, 6/4 und 8/4-Takten, wobei nur der 8/4-Takt das Fenster ganz ausnutzt. Jedem Pattern kann dabei ein anderer Takt (Länge) zugeordnet werden.

Arbeiten mit Patterns

Um nun etwa einen Grundschlag in einem der »Patterns« aufzunehmen, wird mit der Maus auf RECORD geklickt. Das Metronom (in der Ecke rechts unten) wird dann aktiv und durch einen Klickrhythmus (dessen Drumsound mit dem von Taste 9 identisch ist) auch hörbar. Ein weißes Rechteck überstreicht nun das Fenster und zeigt die gerade bearbeitete Stelle an. Nun spielt man mit der Zehnertastaur die gewünschten Trommelschläge; sie erscheinen dann im Fenster als Punkte. Streicht das weiße Rechteck über bereits in einem vorherigen Durchgang aufgenommene Punkte, so werden die Trommelschläge an dieser Stelle abgespielt. Es ist somit zum Beispiel möglich, zuerst einen Baßschlag und dann in weiteren Durchgängen eine Begleitung dazu einzuspielen.

Beim Aufnahmevorgang ist eine weitere Taste des Zehnerblocks mit einer sehr nützlichen Funktion ausgestattet: Wird eine Zifferntaste zusammen mit dem Punkt gedrückt, wird der jeweilige Drumsound automatisch wiederholt. Wie oft der Schlag wiederholt werden soll, ist zwischen 1/4-Note und 1/16-Note variierbar.

Sehr hilfreich für Trommelanfänger ist auch das »Quantizing«. Wird diese Funktion aktiviert, »rundet« der Computer eine bei der Aufnahme gespielte Note (je nach Einstellung des Quantizing-Wertes) auf die nächste Viertel-, Achtel- oder Sechzehntel-Note.

Damit alle Arten von Rhythmen aufgenommen und abgespielt werden können, ist das

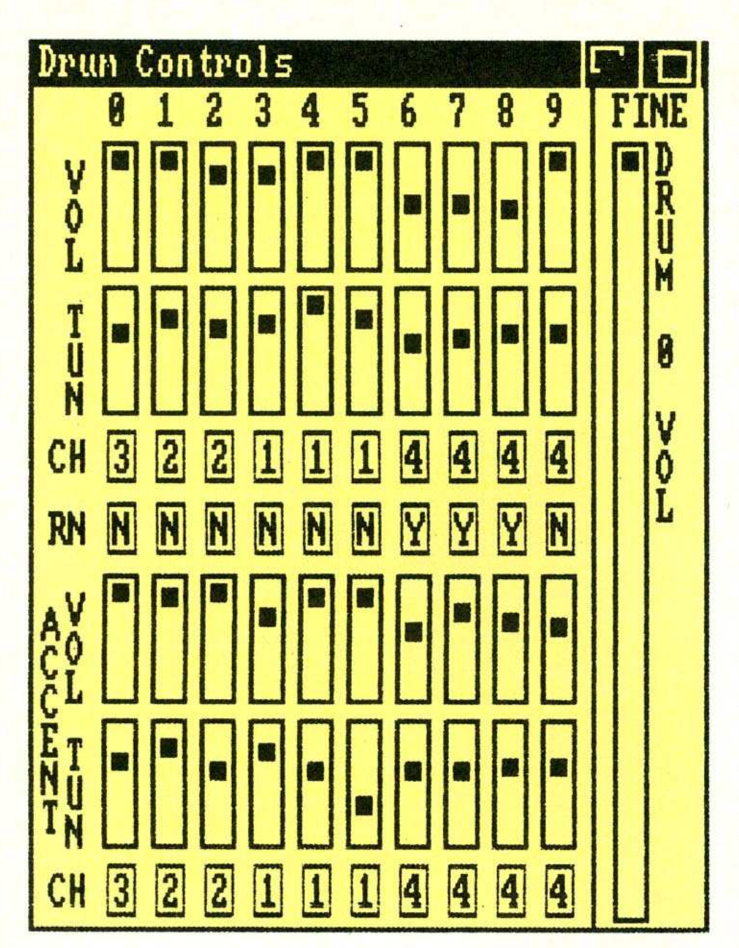


Bild 2. Viele Schieberegler geben die Sound-Kontrolle

Tempo (die Anzahl der Schläge pro Minute) zwischen 40 und 240 einstellbar. Außerdem läßt sich das Programm mit einem am MIDI-Ein/Ausgang angeschlossenen Gerät synchronisieren, wobei die Tempoeingabe natürlich unwirksam wird. Für Aufnahmen von sehr komplizierten Trommelfiguren ist es praktisch, das Tempo auf 40 Schläge pro Minute reduzieren zu können (normal ist etwa 140). Da das weiße Rechteck nun sehr langsam über das Aufnahmefenster streicht, lassen sich auch schnelle Wechsel zwischen den Trommeln leicht spielen. Beim Abspielen wird die Schlagzahl dann einfach wieder erhöht.

Ist im gerade aufgenommenen Pattern (Muster) noch ein Fehler enthalten, kann man es entweder ganz löschen oder mit der Maus bearbeiten. Dabei wird die gewünschte Stelle im Aufnahmefenster angeklickt, an der ein Ton gesetzt, gelöscht oder von normaler auf akzentuierte Spielweise umgeschaltet werden soll. Leider ist es aber nicht ganz einfach, die richtige Stelle im Fenster mit dem Mauszeiger zu treffen, da die einzelnen »Trommelpunkte« eng beieinanderliegen.

Insgesamt ist die Bedienung des Pattern-Systems gut gelungen und leicht zu erlernen. Eine zusätzliche Erleichterung der Arbeit ist die Möglichkeit, ein Pattern in ein anderes zu kopieren. So kann zum Beispiel ein Baßschlag mit mehreren Begleitungen versehen werden.

Um nun aus den einzelnen Patterns ein »Lied« zu machen, werden sie im zweiten Teil des Sequenzers zusammengestellt. Dazu gibt man im Normalfall einfach die Buchstaben der Patterns im Fenster SONGS (rechts oben in Bild 1) ein; die Anzahl der Wiederholungen wird vorangestellt.

»2a3b« würde also bedeuten, daß zuerst Pattern »a« zweimal und dann Pattern »b« dreimal gespielt werden sollen. Das Fenster zur Eingabe der »Songs« kann bis zu 35 Zeichen aufnehmen, was im Normalfall ausreichen dürfte. Mit Hilfe der »X«- und »Y«-Fenster lassen sich zusätzlich noch kleinere Folgen von Patterns als »Unterprogramme« definieren, die dann etwa mit »3x« in das Song-Fenster eingesetzt werden können. Wird auf PLAY geklickt, so wird das Lied abgespielt. Ist die MIDI-Synchronisation aktiviert, wartet das Programm nach dem Klick auf ein externes Startsignal.

Obwohl die auf der Diskette vorhandenen Drumsounds bereits sehr vielfältig sind, erlaubt es Dynamic Drums, den Klang und die Lautstärke jedes einzelnen Drumsounds im Speicher zu verändern. Das geschieht mit Hilfe des »Drum Control«-Fensters (Bild 2), das sich »hinter« der Belegungsanzeige der Zehnertastatur befindet und durch einen Mausklick eingeschaltet wird. In diesem Fenster befinden sich für jede der zehn Tasten Schieberegler mit Tonhöhe und Lautstärke der jeweiligen Trommel — jeweils getrennt für die normale und die akzentuierte Spielweise. Besonders hilfreich ist hierbei der Schieberegler zur Feineinstellung eines Wertes (im Bild 2 rechts), mit dem der gerade aktive Schieberegler noch genauer eingestellt werden kann, was besonders bei der Tonhöhe wichtig ist. Auch während das »Drum Control«-Fenster aktiv ist, können mit dem Zehnerblock die Drumsounds gespielt werden, so daß man jede Einstellung sofort testen kann. Die Schieber beziehen sich auf eine bestimmte Taste des Zehnerblocks und nicht auf einen bestimmten Drumsound. Die eingestellten Werte werden also nur auf Diskette gespeichert, wenn der Menüpunkt »Save Drumkit« verwendet wird.

Starke Patterns

Wer viel mit den Schieberegler experimentiert, wird feststellen, daß eine Verstellung der Tonhöhe manche Drumsounds wirklich nur in der Höhe verändert, andere aber vollkommen anders klingen läßt. Besonders interessant wird diese Einstellmöglichkeit natürlich etwa bei den Gitarrentönen, mit denen man so auch kleine Melodien oder Baßrhythmen spielen kann. Um das zu tun, muß man allerdings einen Drumsound mehrmals von der Diskette laden, da ein Menüpunkt zum Kopieren von Drumsounds fehlt.

Sehr wichtig ist auch die im Drum-Control-Fenster vorhandene Einstellung des Tonkanals, mit dem der jeweilige Drumsound gespielt wird. Da der Amiga nur vier Tonkanäle hat, ist es nicht möglich, zehn Töne gleichzeitig zu spielen. Damit sich die Drums nun nicht gegenseitig beim Ausklingen stören, wird man Baßdrum und High-Hat auf zwei verschiedene Tonkanäle setzen.

Um die Computertrommel etwas realistischer klingen zu lassen, ist über das Drum-Control-Fenster ein Zufallsgenerator aktivierbar. Ist er eingeschaltet, so wird beim Spielen eines Tones ständig zwischen der bei »normal« und bei »akzentuiert« eingestellten Tonhöhe (und Lautstärke) zufällig variiert, so daß jeder Ton eine etwas von seinem Vorgänger unterschiedliche Höhe hat.

Ein besonderes Augenmerk verdient bei Dynamic Drums die Anleitung. Auf acht Seiten beschreibt sie in knappen, aber gut verständlichen englischen Worten die Bedienung des Programms. Gut gelungen ist eine dem Programm beigelegte Kassette, auf der ein freundlicher Amerikaner die Aufnahme eines Drum-»Liedes« von Anfang an erklärt, wobei man sich sofort anhand der auf der Kassette hörbaren Beispiele überprüfen kann. Diese Art der hörbaren Anleitung sollte bei Musikprogrammen Schule machen.

Insgesamt ist Dynamic Drums ein Programm, das trotz einiger Schwächen als Ersatz für einen Drumcomputer gut geeignet ist. Einer professionellen Verwendung könnte höchstens die Amiga-Samplingrate mit nur 8 Bit einen Strich durch die Rechnung machen. (Andreas Lietz/jk)

AMIGA-WERTUNG

Software: Dynamic Drums

9,0 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	x	x	x	x	x	
Dokumentation	x	x	x	x		
Bedienung	x	x	x	x	x	
Erlernbarkeit	x	x	x	x	x	
Leistung	x	x	x	x	x	

Fazit: Dynamic Drums ist ein Drumprogramm mit Sequenzer, das durch die gute Bedienung und die MIDI-Synchronisation auch für Musikprofis geeignet ist. Die vielen mitgelieferten Drumsounds machen das Programm vielseitig einsetzbar.

Positiv: Grafische Bearbeitung von Drumrhythmen; viele Sounds mitgeliefert; vielfältige Änderungsmöglichkeiten für Sounds.

Negativ: Kopierschutz; Laden von Drumkits dauert zu lang; beim Laden von fremden Soundfiles Absturz möglich.

DATEN

Produkt: Dynamic Drums
Preis: 139 Mark
Hersteller: New Wave Software
Anbieter: Intelligent Memory, Basaltstr. 58, 6000 Frankfurt, Tel. 707 11 02

Digitalisierung in Perfektion

AMIGA test

Mit der Sampling-Software »Audio Master« hat das Software-Haus Aegis wieder ein neues Produkt auf den Markt gebracht. Kann dieses Programm den hohen Standard seiner Vorgänger halten?



Bild 1. Der »Audio Master« ist die neue Sampling-Software vom bekannten amerikanischen Software-Haus Aegis

Bei vielen auf dem Markt befindlichen Sound-Digitalisierern hapert es noch mit komfortabler Software zum Editieren der aufgenommenen Sequenzen. In diese Marktlücke tritt der Audio Master, ein Programm, das als Erweiterung zu schon vorhandenen Digitizern gedacht ist.

Audio Master ist deswegen nur ein Hilfsprogramm zur Digitalisierung von Tönen und keineswegs ein komplettes Sampling-Paket, da die Hardware zur Tondigitalisierung fehlt. Angeschlossen werden die meisten der am Parallel- oder Joystickport arbeitenden Digitizer von anderen Firmen.

Audio Master wird mit einem 72seitigen, englischen Handbuch ausgeliefert, das alle Funktionen des Programmes beschreibt. Trotzdem wäre eine weitergehende Definition bestimmter Ausdrücke wünschenswert gewesen; der Anfänger könnte von diesem Handbuch verwirrt werden.

Zuerst werden Sie in die Grundsätze der Aufzeichnung von Schallwellen und deren Digitalisierung eingeführt, so daß sich das erste Kapitel etwas trocken gestaltet. Kurz darauf geht es aber auch schon los, mit den handfesten Tips und Tricks zur Bedienung.

Durch die Beschränkung auf wenige Menüpunkte, die jedoch meist ein eigenes Fenster aufrufen, wurde die Bedienung des Audio Master sehr übersichtlich gestaltet. Während den ersten Arbeiten fällt bei diesen Extrafenstern schon angenehm auf, daß trotz deren Vorhandensein gleichzeitig im Hauptfenster von Audio Master weitergearbeitet und zum Beispiel die Lautstärke verändert werden kann. Die einzige Ausnahme bildet hier das Digitalisierungsfenster, da beim Ansprechen des Digitizers unter anderem auch die Mausfunktion abgestellt wird.

Überhaupt sollten Sie einige Ratschläge des Handbuches genau befolgen, da sonst mit der Zeit Datenverluste und Gurus garantiert sind. Dies betrifft vor allem die Hinweise vor einer Digitalisierung erst auf das Erlöschen sämtlicher Floppy-LEDs zu warten sowie während des Digitalisierungsvorganges keine Disketten aus den Laufwerken zu nehmen oder hineinzustecken.

Doch erläutern wir zuerst einmal einen normalen Digitalisierungsvorgang.

Abgleich mit Hindernissen

Nachdem Sie Ihren Mono-Digitizer angeschlossen (bei den Stereoversionen wird nur ein Kanal digitalisiert) und das Hauptprogramm geladen haben, kann durch die Anwahl des Sample-Menüs ein weiteres Fenster aufgerufen werden. In der jetzt vorhanden Menüzeile muß der verwendete Digitizertyp eingestellt und daraufhin abgeglichen werden. Bei Digitizern am Parallel-Port geht dies direkt im Sample-Menü, bei den Joystick-Varianten muß ein eigenes Programm von der Hauptdiskette geladen werden. Dazu müssen Sie allerdings den Audio Master wieder verlassen, da er nur eingeschränkt multitaskingfähig ist. Beispielsweise wird kein eigener Bildschirm, sondern nur ein NTSC-großes Fenster geöffnet, das sich nicht verschieben oder vergrößern läßt. Da auch eine Verkleinerung nicht möglich ist, können Sie, während Audio Master aktiv ist, keine Diskettensymbole anklicken, um andere Programme zu laden. Wie oben schon erwähnt, übernimmt Audio Master den Amiga bei der Digitalisierung vollständig, das heißt alle Interrupts und Tasks werden kurzzeitig abgeschaltet. Dies könnte bei manchen Programmen zu Timingproblemen führen, aber eben deswegen läßt Audio Master nur eingeschränktes Multitasking zu.

Ist der Digitizer auf den Nullpunkt abgeglichen, sollte die gewünschte Länge der Digitalisierung angegeben werden. Hier kann zwischen null und dem größten zusammenhängenden (!), nicht belegten Speicherblock fast jeder beliebige Wert angewählt werden.

Schalten Sie nun auf den Monitorbetrieb, hören Sie das am Digitizer anliegende Signal durch den Lautsprecher. Mit »Sample LO« oder »Sample HI« können Sie schließlich in verschiedenen Qualitäten digitalisieren. Leider sind hier nur feste Abtastraten von 8363 sps (samples per second) und 19886 sps (parallel) beziehungsweise 14914 sps (Joystick) möglich. Durch die Möglichkeit, nachträglich die Abtastrate stufenlos zu verändern, fällt das nicht so ins Gewicht, zumal die Frequenz von 8363 sps genau der von den meisten Musikprogrammen erwartete Samplingwert eines Instrumentes ist.

Während des Monitorbetriebs sehen Sie auf dem Bildschirm eine Art Oszillographenbild, das die jeweilige Frequenz angibt und nach dem die benötigte Lautstärke exakt eingeregelt werden kann.

Der jetzt erfaßte Klang wird auf dem Bildschirm grafisch dargestellt und kann von nun an frei verändert werden. Neben den schon bekannten Funktionen zum Ausschneiden, Kopieren und Einfügen eines Abschnittes bietet Audio Master jedoch noch einiges mehr an Spezialeffekten.

So kann mittels eines Knopfdruckes sofort die nächste Nullstelle aufgefunden werden, was zur Vermeidung von Knackstönen von äußerster Wichtigkeit ist.

Doch nun zu den Spezialfunktionen von Audio Master. Im Echo-Menü können Sie einen vorher angewählten Bereich per Echo wiederholen lassen. Dabei bleibt natürlich die Anzahl der Wiederholungen des Echos beliebig einstellbar. Aber auch die Echo-Lautstärke und die Geschwin-

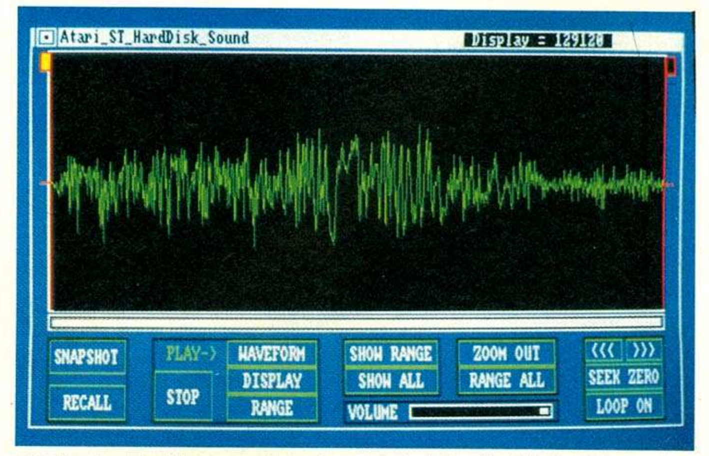


Bild 2. Im Bedienungsfenster zeigt sich die Wellenform jedes einzelnen Klangs als grafische Schwingung

digkeit des Echos kann frei verändert werden. Dabei wird das Echo auf alle schon vorhandenen Geräusche aufgetragen, wirkt also sehr realistisch.

Die Umkehrung eines Bereiches kann verblüffte Gesichter bei Zuschauern hervorrufen, wenn diese versuchen ihren Namen rückwärts in ein Mikrophon zu sagen und dieser dann nach einer Umkehrung richtigherum abgespielt wird.

Großen Rechenaufwand haben die folgenden Funktionen zu bewältigen. Sie benötigen deshalb oft einige Minuten intensiver Rechenzeit, um zu einem Ergebnis zu gelangen.

Tolle Effekte

Als erstes wäre da die »Mix Wave«-Funktion zu erwähnen, die das Mischen zweier Wellenformen unter Berücksichtigung verschiedener Einstellungen, wie zum Beispiel der Teillautstärke oder der Geschwindigkeit, ermöglicht.

Mit dem »Change Volume«-Menü können Sie nachträglich jede Stelle Ihres Klanges lauter oder leiser erklingen lassen. Sogar weiche Übergänge in den Lautstärken lassen sich mit dieser Funktion bewerkstelligen. Das wohl umfangreichste Menü bietet »Tune Waveform«. Hier können sämtliche, den ganzen Klang betreffende, Parameter verändert werden. So ist ein »Resample« mit einer anderen Frequenz genauso möglich, wie die Definition der Oktaven und der Höhe einzelner Töne. Zum besseren Vergleich kann neben dem eigentlichen Klang noch ein, auch frei veränderbarer, Referenzklang ertönen. Dieser Klang kann das Feinstimmen erheblich erleichtern. Dabei ist es auch der einzige Punkt, an dem Audio Master mehr als einen Audiokanal des Amiga anspricht. Leider wurden keinerlei Funktionen zur Mischung von Klängen auf verschiedenen Kanälen implementiert, wie man das von einigen anderen Programmen her kennt.

Bei tiefen Tönen aufkommende Verzerrungen können mittels eines Software-Low-Pass-Filters weitestgehend eliminiert werden.

Weitere Funktionen von Audio Master beziehen sich auf die Vorauswahl, ob bei jedem Kommando, das größere Veränderungen nach sich zieht, eine Abfrage gemacht werden soll, oder ob ausgeschnittene Teile grundsätzlich in den Puffer übergeben werden sollen.

Da Audio Master keine UNDO-Funktion besitzt, half man sich hier anders. Mit »Snapshot« (nicht zu verwechseln mit der gleichnamigen Workbench-Funktion) kann der momentane Zustand auf ein vorher definiertes Laufwerk abgelegt und mit »Recall« wieder zurückgerufen werden. Leider kann das »Snapshot-Device« nur RAM:, DH0:, DF0: oder DF1: sein, was aber in den meisten Fällen ausreichen dürfte.

Am Ende des Handbuches findet der Profi noch eine Menge Tips und Tricks, unter anderem vom schon fast legendären Pepper Brown, der für viele Spitzensongs auf dem Amiga verantwortlich zeichnet.

Audio Master ist ein ausgefeilter Editor für digitale Klänge, der mit vielen neuartigen Funktionen überrascht. Für jeden ist etwas dabei, egal, ob gut klingende Instrumente für ein Musikprogramm oder tolle Knalleffekte für das neueste Spiel geschaffen werden sollen.

(Ottmar Röhrig/jk)

AMIGA-WERTUNG

Software: Audio Master

9,0 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	x	x	x	x	x	
Dokumentation	x	x	x	x	x	
Bedienung	x	x	x	x	x	
Erlernbarkeit	x	x	x	x		
Leistung	x	x	x	x		

Fazit: Der Audio Master bietet Soundsampling-Software mit vielen Extra-Funktionen, die man bisher bei anderen Programmen vermißt hat. Trotz einiger nicht erfüllter Wünsche ergibt sich ein gutes Gesamtbild.

Positiv: viele Effekte; Anpassung an verschiedene Digitizertypen.

Negativ: nur Monodigitalisierungen; kein eigener Bildschirm; eingeschränkt multitaskingfähig.

DATEN

Produkt: Audio Master
Preis: 98 Mark
Hersteller: Aegis Development
Anbieter: Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081

Marktübersicht
Soft- und Hardware für Musik

Im Sektor Musik tut sich auf dem Amiga einiges. Um Ihnen einen Überblick über die verschiedenen Bereiche und Produkte zu geben, haben wir alle wichtigen Daten in einer Übersicht zusammengefaßt. Zum Abschluß ergänzt eine ausführliche Liste mit den genauen Adressen der wichtigsten Hersteller die Aufstellung.

Alles, was auf dem Amiga mit Musik zu tun hat, läßt sich grundsätzlich in die Bereiche Soft- und Hardware unterteilen. Dies äußert sich dann auch in der Struktur der Marktübersicht, die zuerst den weitaus größeren Teil an Software-Produkten aufführt und danach die beiden bisher bestehenden Einteilungen für Musik-Hardware vornimmt. An Hardware gibt es zur Zeit die verschiedenen Sampler zur Digitalisierung von Tönen und Geräuschen und die MIDI-Interfaces.

Beide Arten von Hardware werden in unterschiedlichen Versionen für den Amiga 1000 und Amiga 500 beziehungsweise Amiga 2000 angeboten.

Vor einem Kauf sollte man sich vergewissern, daß man später auch genau die Version geliefert bekommt, die man gebrauchen kann. Dies liegt in der unterschiedlichen Pin-Belegung des seriellen Ports an der Rückseite der verschiedenen Amiga-Modelle begründet. Im Einsteigerteil dieser AMIGA-Ausgabe wurde schon etwas ausführlicher auf die Grundlagen von Samplern eingegangen, was durchaus dazu ausreichen dürfte, die verschiedenen Modelle einzuschätzen und ein dem eigenen Bedarf entsprechendes Gerät herauszufinden.

Bei den MIDI-Interfaces muß man sich über den eigenen Anspruch und die Ausstattung an anderen MIDI-Gerätschaften im klaren sein. Einige Interfaces sind sehr einfach aufgebaut und bieten nur einen Aus- und Eingang. Andere Modelle haben schon die MIDI-Thru-Buchsen integriert und verwenden zusätzlich noch Umschalter für die einzelnen Ausgänge. Damit kann der Aufbau einer MIDI-Anlage rein hardwaremäßig vom Interface beeinflußt werden. Die Größe des Interfaces und die damit verbundene Preisklasse muß jeder letztendlich selbst bestimmen. (jk)

Musikprogramme (notenorientiert)	Hersteller	Preis in Mark	Anbieter
Deluxe Music Construction Set	Electronic Arts,	193-249	A,B,F,I,J,K,L,P,Q
Sonix	Aegis	147-289	A,B,E,F,I,J,K,N,P,Q
Musicstudio	Activision	97-99	A,B,J,Q
Music X	Micro Illusion	537-558	B,Q
Musikprogramme (nicht notenorientiert)			
Instant Music	Electronic Arts	69-198	A,B,F,I,J,N,Q
Music Mouse		240	M
Drumprogramme			
Drum Studio	Golden Games	79	E,G
Dynamic Drums	New Wave	139	A,I
Turbo Drummer	Roßmöller	149	O
Soundsampler			
Audio Master	Aegis	98	A,B,J
DeLuxe Sampler	Hagenau	198-228	H
Futuresound	Applied Visions	337-395	A,B,J,P
Perfect Sound	SunRize Ind.	169-225	A,B,P,R
Sounddigitizer		149	S
Soundsampler	Compware	198	D
Sound Sampler		219	J
Soundscape (inkl. Pro Midi)	Mimetics	288-349	A,B,J,M,P
MIDI-Programme			
Pro Midi Studio (inkl. Soundscape)	Mimetics	288-349	A,B,J,M,P
Synthesizer			
Synthia	Other Guys	248	B
Musik- und Sound-Zusatzdisketten			
Effect Creator	Roßmöller	99	O
IFF Music Data	Party Sound	je 49	A
Instant Music Data	Electronic Arts	68	A,B,F,I,J
Sonix Dreams 1,2,3,4	Atlantis	je 48	B
Wavetable Sounds	Wavetable Tech	je 69	A,I
MIDI-Interfaces			
ECE Midi	Midi Interfaces	116-188	B,P,R
GOL Midi-Interface		153	P
Midi-Interface	Compware	98	D
Midi-Interface	Hagenau	98	H
Midi-Interface	Mimetics	120	M
Midi-Interface		89	C
Midi-Interface		99	J
MIM Midi-Interface		95	P

Anbieter (kein Anspruch auf Vollständigkeit):

A: Amigaland, Hohenwaldstr. 26, 6374 Steinbach, Tel. 06171/71846
B: Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081
C: CAS, Sprendlinger Landstr. 71, 6050 Offenbach, Tel. 069/842013
D: Compware, Schlägel & Eisen Str. 9, 4352 Herten, Tel. 02366/55891
E: CSJ, An der Tiefenriede 27, 3000 Hannover, Tel. 0511/886383
F: DTM, Poststr. 25, 6200 Wiesbaden, Tel. 06121/560084
G: Golden Games, Berliner Platz 5, 4050 M'Gladbach, Tel. 02161/10735
H: Hagenau Computer, Münstererstr. 202, 4700 Hamm, Tel. 02381/673165
I: Intelligent Memory, Basaltstr. 58, 6000 Frankfurt, Tel. 069/7071102
J: Jumbo Soft, Horemannstr. 2, 8000 München, Tel. 089/1234065
K: Medien Center, Wermingser Str. 45, 5860 Iserlohn, Tel. 02371/24599
L: Markt & Technik, Hans-Pinsel-Str. 2, 8013 Haar, Tel. 089/4613-0
M: MEV Midi Soft, Karl-Hromadnik Str. 3, 8000 München, Tel. 089/835031
N: Philgerma, Barerstr. 32, 8000 München, Tel. 089/395551
O: Roßmöller, Maxstraße 50-52, 5300 Bonn, Tel. 0228/650212
P: Softwareland, Franklinstr. 27, CH-8050 Zürich, Tel. 0041-1-3115959
Q: Soyka, Hattinger Str. 685, 4630 Bochum 5, Tel. 0234/411913
R: Video Loft, Fiedlerstr. 22-32, 3500 Kassel, Tel. 0561/873399
S: Wolf, Deipe Stegge 187, 4420 Coesfeld, Tel. 02541/2874

Ein Ball schlägt sich durch

AMIGA test **Ausgerechnet eines der allerersten Videospiele erlebt in diesen Tagen eine wirklich sensationelle Wiedergeburt. Das Angebot an »Breakout«-Varianten ist sehr vielfältig. Zwei der besten Versionen treten zum Vergleichstest an.**

Kaum ein Monat vergeht, ohne daß eine neue »Breakout/Arkanoid«-Variante für den Amiga veröffentlicht wird. Zwei neue Programme streiten sich um die Vormachtstellung in diesem hart umkämpften Spielgenre. Neben »Impact« und »Detonator« gibt es zwar noch andere Arkanoid-Versionen, doch diese sind älter und deutlich schlechter.

Bei Impact, das aus England kommt, gibt es 128 Bilder, von denen 80 fix und fertig sind. Die restlichen 48 kann man sich mit einem Editor selber gestalten. Beim Steineabräumen fällt manchmal ein gelber Bonusgegenstand zu Boden. Indem man diese Symbole sammelt, sind verschiedene Extras zu aktivieren. Für jedes Extra, das per Druck auf die rechte Maustaste angewählt wird, brauchen Sie eine bestimmte Anzahl von Symbolen.

Die Auswirkungen der Extras sind recht unterschiedlich. Eines bremst den Ball, ein anderes ruft zwei Zusatzbälle ins Leben, ja sogar Klebstoff für den Schläger ist vorhanden und unsichtbare Ziegel können sichtbar gemacht werden.

Wem das Steineabräumen zu langwierig ist, der sollte sich den Laser besorgen, mit dem sich die Ziegel einfach wegschießen lassen. In manchen Situationen ist auch die Bombe sehr nützlich. Sie zerstört alle Sprites, die sich gerade auf dem Spielfeld tummeln und die Flugbahn des Balles beeinflussen (Bild 1).

Nach jeweils zehn Runden erhält man ein Paßwort, so daß nicht immer wieder von vorne begonnen werden muß. Die Grafik von Impact ist einfach aber zweckmäßig. Digitalisierte Soundeffekte sind ebenfalls mit von der Partie. Dank des cleveren Extra-Systems fließt eine taktische Note in den Spielverlauf ein. Der Schläger wird mit der Maus gesteuert. Ein Level-Editor sorgt für Abwechslung, auch wenn man alle 80 eingebauten Levels schon geschafft hat.

Bild 1. »Impact« erfordert auch einiges an Strategie durch das ausgefeilte System der Extrapunkte

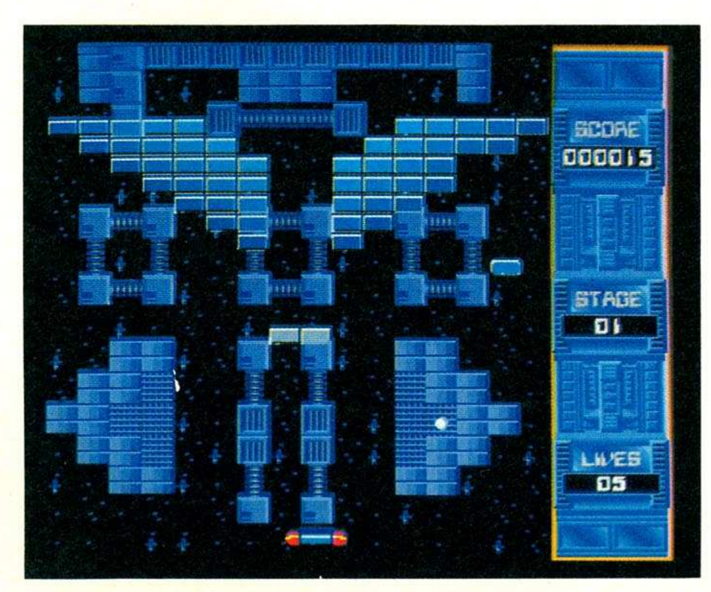


Bild 2. Hervorragender Sound und schnelle Grafik in PAL-Auflösung zeichnen den »Detonator« aus

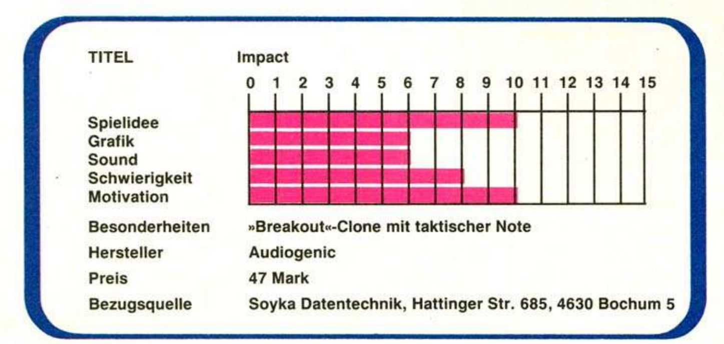
Ein Produkt aus Deutschland versucht Impact den Rang abzulaufen. Detonator lehnt sich stark an den Spielautomatenhit Arkanoid an. Im Gegensatz zu Impact werden die Extras in dem Moment aktiviert, wenn man die verschiedenen Symbole auffängt. Man kann den Ball verlangsamen, sich Klebstoff besorgen, zwei Zusatzbälle herbeizaubern, den Schläger vergrößern und ein Zusatzleben ergattern. Als besonderes Zuckerl gibt es das Relax-Extra. Fängt man dieses Symbol auf, übernimmt der Computer für kurze Zeit die Steuerung des Schlägers.

Detonator bietet 50 Level und eine speicherbare High-Score-Liste. Zur Steuerung des Schlägers sind Joystick oder Maus erlaubt. Erfreulicherweise nutzt Detonator die PAL-Auflösung aus. Somit entfällt der Cinemascope-Streifen am unteren Bildschirmrand. (Bild 2). Sowohl digitalisierte Soundeffekte als auch programmierte Musik sind während des Spiels zu hören.

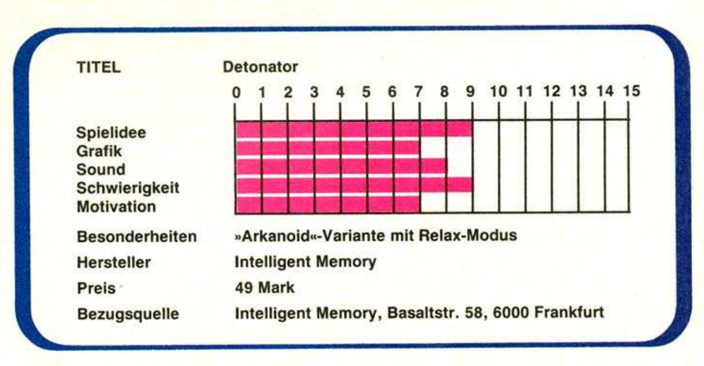
Im direkten Vergleich schneidet Impact etwas besser als Detonator ab. Für Detonator sprechen zwar schönere Grafik und spektakulärere Soundeffekte, dafür spielt sich Impact deutlich besser. Hinzu kommt das intelligente Extra- und Paßwort-System, das den Spielreiz anstachelt. Ein besonderes Lob verdient der gut gemachte Level-Editor bei Impact. All dies fehlt bei Detonator leider gänzlich.

Fans dieses Spielgenres sind mit beiden Produkten recht gut bedient. Wer Detonator schon besitzt, der sollte sich Impact trotzdem noch einmal anschauen.

(Martin Gaksch/hl/jk)



TITEL	Impact
Besonderheiten	»Breakout«-Clone mit taktischer Note
Hersteller	Audiogenic
Preis	47 Mark
Bezugsquelle	Soyka Datentechnik, Hattinger Str. 685, 4630 Bochum 5



TITEL	Detonator
Besonderheiten	»Arkanoid«-Variante mit Relax-Modus
Hersteller	Intelligent Memory
Preis	49 Mark
Bezugsquelle	Intelligent Memory, Basaltstr. 58, 6000 Frankfurt



Ein AMIGA kommt selten allein!
Commodore SIDECAR gehört dazu!
Macht Ihren AMIGA 1000 IBM-PC kompatibel!
Mit PC-Board, 5.25" Floppy-Disk, 3 Erweiterungs-Slots für Steckkarten oder Hard-disk-Card.
(Eine Harddisk kann sowohl unter MS-DOS als auch unter AMIGA-DOS benutzt werden.)
Personal Computer
PC 10
PC 20
DOS
Microsoft® MS™-DOS
Commodore
Commodore PC 10 PC 20
DOS
Im Lieferumfang enthalten:
MS-DOS 3.20 auf 5.25" Disketten
AMIGA Kickstart 1.2 auf 3.5" Diskette
AMIGA Workbench 1.2 auf 3.5" Diskette
Preissenkung!
Alter VOBIS-Preis
[redacted].-
jetzt nur noch
999.-
AMIGA
Das komplette AMIGA-Angebot bei VOBIS:
AMIGA 500 incl. HIGHSCREEN-Stereo-Farbmonitor 1599.-
A 501 Speichererweiterung dazu 275.-
A 520 HF-TV-Modulator dazu 59.-
AMIGA 1000 incl. GrafiCraft, TextCraft und MS-Transformer Disk 1695.-
SIDECAR PC-Aufrüstung dazu 999.-
AMIGA 2000 incl. HIGHSCREEN-Stereo-Farbmonitor 2795.-
2. Einbaulaufwerk 3.5" 348.-
PC-Board incl. 5.25"-Floppy 998.-
2. Laufwerk extern 3.5" für AMIGA 500, 1000, 2000 345.-
Jetzt neue Filialen in Krefeld + Aachen!
kompetent + preiswert
WICHTIG! Herstellerbedingte Lieferzeiten! Aufgrund erhöhter Nachfrage ist nicht immer alles sofort lieferbar!
VOBIS
Deutschlands umsatzgrößter Microcomputer-Spezialist
VERSAND-ZENTRALE:
Postfach 1778
Rotter Bruch 32-34
5100 AACHEN
☎ 0241/500081
Tx 832389vobis d
FILIALEN:
1000 BERLIN 30
Kurfürstenstr. 101 · 030/2 13 94 80
2000 HAMBURG
Krohnskamp 15 · 040/2 79 46 76
2300 KIEL
Sophienblatt 74-78 · 0431/67 86 22
2800 BREMEN
Violenstraße 37 · 0421/32 04 20
3000 HANNOVER
Berliner Allee 47 · 0511/81 65 71
4000 DÜSSELDORF
Wielandstr. 21 · 0211/35 99 64
4150 KREFELD
Ostwall 92 · 02151/80 07 93
4300 ESSEN
Huyssenallee 3 · 0201/23 17 74
4600 DORTMUND
Hamburger Str. 110 · 0231/57 30 72
4800 BIELEFELD
Herforder Str. 106 · 0521/6 38 78
5000 KÖLN
Mathiasstr. 24-26 · 0221/24 86 42
2x 5100 AACHEN
Viktoriastr. 74 · 0241/54 31 00
Großkölnstr. 60 · 0241/2 44 94
(gemeinsam mit Foto PORST)
6000 FRANKFURT
Frankenallee 207/209 · 069/73 40 49
7000 STUTTGART
Marienstr. 11-13 · 0711/60 63 36
7500 KARLSRUHE
Kriegsstr. 27/29 (am BGH) 0721/37 82 68
7750 KONSTANZ
Kreuzlinger Str. 18 · 07531/1 55 60
8000 MÜNCHEN
Aberlestr. 3 · 089/77 21 10
8500 NÜRNBERG
Vordere Ledergasse 8 · 0911/23 29 95
8900 AUGSBURG
Jakoberstr. 16 · 0821/152349

TURBO-POWER: Neue Welten erobern

Turbo-Geschwindigkeit macht den Amiga auch in der an schnelle Computer gewöhnten Industrie salonfähig. Die hohe Rechenleistung kann mit den bisher eingesetzten Systemen wie zum Beispiel einer Sun-Workstation oder einer VAX 11/780 konkurrieren. Dennoch die Frage: Lohnt sich die Investition?

Der schnelle 68020-Prozessor von Motorola, ein zusätzlicher 68881-Arithmetikprozessor, schnellere RAM-Chips und eine gesteigerte Taktfrequenz von bis zu 16 MHz für den Hauptprozessor beschleunigen den Amiga wie eine Rakete. Sowohl die meisten für den Amiga erhältlichen Programme laufen auf dem Turbo, als auch speziell für den neuen Prozessor und seinen Partner konzipierte Software. Wie stark die Geschwindigkeit gesteigert wird, hängt von der Anwendung ab.

Die erste Beschleunigung resultiert aus der Verwendung der höheren Taktfrequenz. 7,14 MHz takten den 68000 im Amiga. Der 68020 auf der von CSA gefertigten Turbo-Karte verträgt 14,32 MHz. Der Arithmetik-Prozessor kann sogar bis zu 25 MHz verkraften. Wenn Sie den Turbo mit einer normalen Workbench starten, läuft bereits die gesamte Software bis zu dreimal schneller. Es profitieren besonders Programme, die mit Langworten und Registervariablen operieren. Probleme tauchen einzig und allein bei Programmen auf, die den »MOVE.SR«-Befehl verwenden. GIT liefert aber Programme mit, die dieses Übel in fast allen Fällen beseitigen.

Die erste Stufe

Ein C-Programm zur Berechnung von Primzahlen nach dem Sieb des Erathostenes, das mit 32 Bit langen Ganzzahlen rechnet, arbeitet immerhin 2,7mal schneller (siehe Tabelle).

Die RAM-Chips und der 68000-Prozessor im Amiga besitzen einen 16 Bit breiten Datenbus. Langworte werden in zwei Worten übertragen. Da der 68020 einen 32 Bit breiten Datenbus besitzt, können Daten schneller transportiert werden, wenn auch die Speicher-Chips in der Lage sind, 32 Bit parallel zu verarbeiten. Hierzu gibt es eine spezielle RAM-Karte für den Turbo-Amiga. Die Routine ADDRAM im C-Ordner der Turbo-Workbench initialisiert diesen Speicher. Jetzt kann der Computer mit 32 Bit Daten in den speziellen Speicher pumpen und aus den Speicherstellen holen. Aus der Öffnung des 16-Bit-Engpasses ergibt sich eine zusätzliche Steigerung des Rechentempos um bis zu 100 Prozent. Damit läuft der Turbo bereits drei- bis fünfmal so schnell wie ein normaler Amiga 2000: Basic-, C- oder Assemblerprogramme,

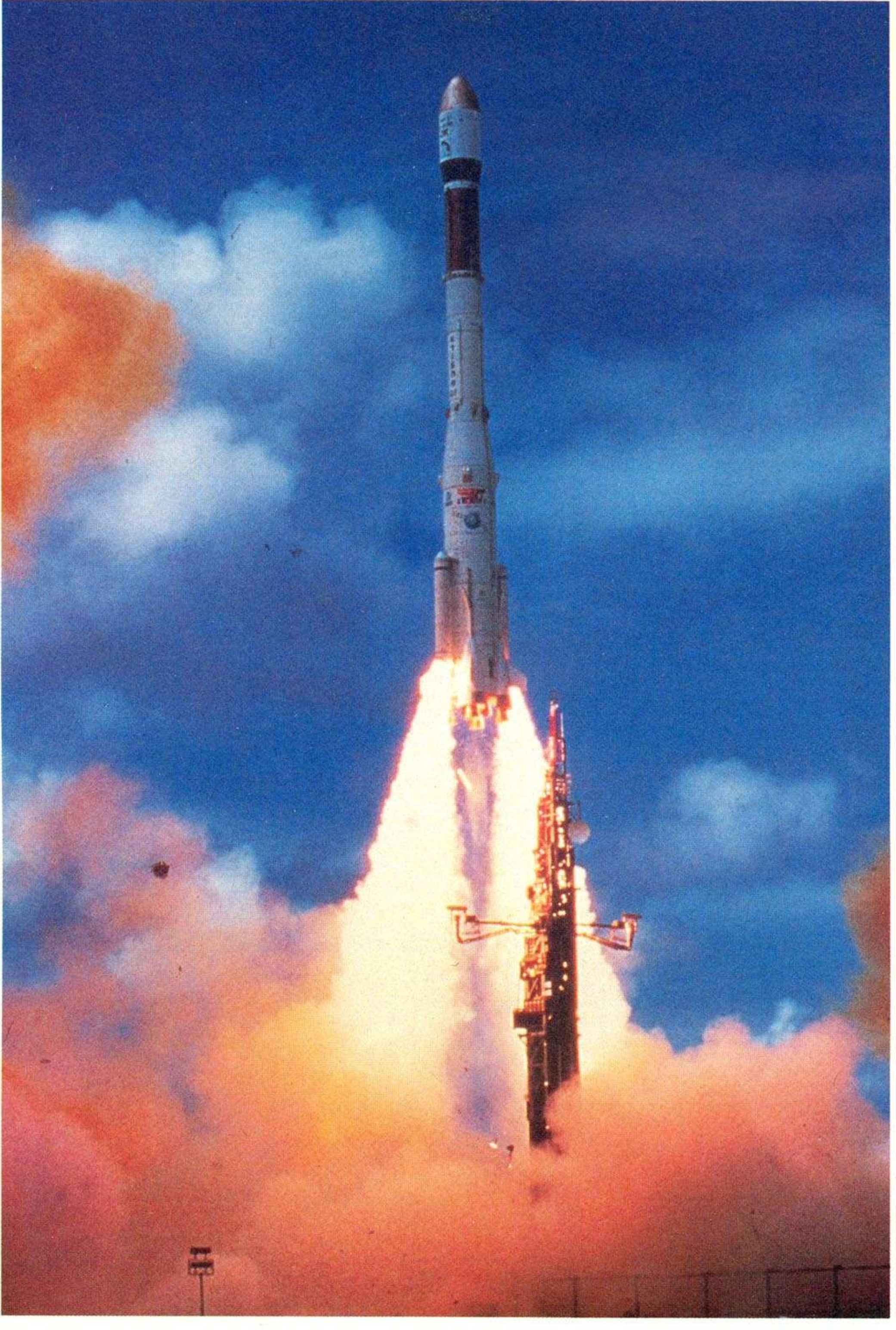


Foto: MBB ERNO

Compiler, Anwenderprogramme oder Spiele, alles arbeitet schneller. Allerdings verträgt sich nicht die gesamte Software mit dem aufgrüsteten Amiga. Einige Spielprogramme, etwa Garrison, laufen nicht. Aber der Turbo ist auch kein Computer zum Spielen.

Das bisher erreichte Tempo schafft der Turbo-Amiga quasi aus dem Stand. Noch rasanter rechnet er, wenn er den 68881 Prozessor einsetzt. Dies erfordert spezielle Programme, die den Coprozessor ansprechen. Spezielle Mandelbrotprogramme werden relativ zum Amiga 2000 um den Faktor 6 bis 8 beschleunigt. Der direkte Vergleich eines Amiga 2000 mit dem Turbo-Amiga, dessen Fast-RAM bereits installiert wurde, zeigt, wie schnell der Turbo rechnet (Bild 1).

Hyperspace: Der Nachbrenner

Für Programmierer ist gesorgt. Sowohl Aztec-C Version 3.4, AC-Fortran 77/020 von Absoft und der Assembler von Quello erlauben die Unterstützung des Coprozessors für Fließkommaberechnungen.

Besonders Aztec-C ist ein interessantes Hilfsmittel für die Entwicklung von Turbo-Software. Alle auf dem Amiga mit diesem Compiler geschriebenen Quelldateien müssen nur mit der Option +F8 compiliert werden. Beim Linken muß der Entwickler die speziellen Mathematik-Bibliotheken auf der vierten Diskette des Aztek-Pakets einbinden.

Einige kommerzielle Software wurde bereits an den Turbo angepaßt. »Flight II« ist eines dieser Programme. Schnell wie ein Blitz gleitet das Flugzeug in der Turboversion über die Landschaft. »Logistix« und »SPlan« befinden sich in der Vorbereitung. Ein weiteres, bereits angepaßtes Programm ist »Sculpt 3D«. Die Beschleunigung durch den 68881 macht sich vor allem in rechenintensiven Programmen bemerkbar, die mit Fließkommazahlen operieren. Gerade bei der Erzeugung von dreidimensionalen Bildern muß der Computer viel rechnen. Dauert die Berechnung eines Objekts (Bild 2) mit dem Amiga noch 2 Stunden und 34 Minuten, so schafft der Turbo dieselbe Aufgabe spielend in 24 Minuten.

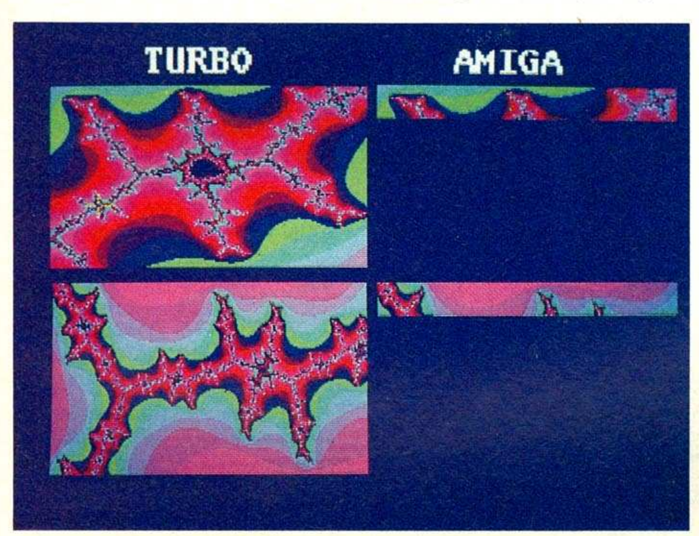


Bild 1. Im direkten Vergleich zeigt der Turbo seine Leistung

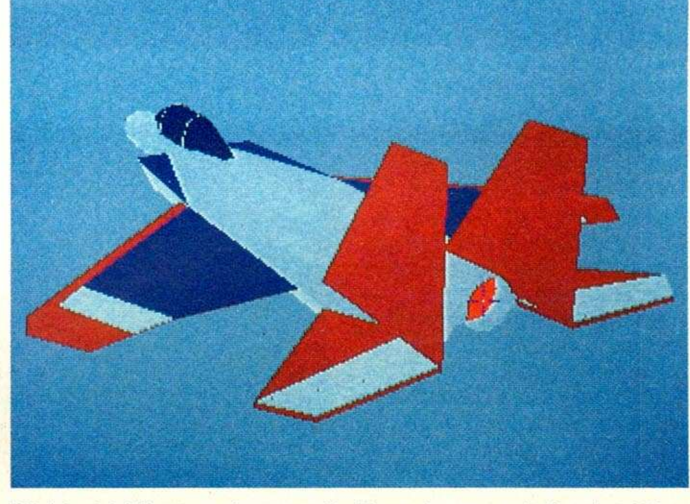


Bild 2. 24 Minuten dauerte die Berechnung mit Sculpt 3D

Die Tabelle zeigt Ihnen weiter Beschleunigungsfaktoren des Turbos. Die Tests der mathematischen Funktionen wurden mit Aztec-C durchgeführt. Für jede Operation wurde die Dauer von 100 000 Berechnungen gemessen. Die Relation zur Laufzeit des Amiga 2000 zeigt den Beschleunigungsfaktor. Unter der Berücksichtigung der Laufzeiten für eine Leerschleife ergeben sich sogar 35fache Temposteigerungen. Durch geschickte Programmierung lassen sich einige Geschwindigkeiten noch weiter steigern. Ähnlich wie die Definition von Registervariablen in C unterstützt Aztec-C auch bis zu vier Variablen in den Registern des schnellen 68881-Coprozessors.

Eine Frage, die sicher jeden interessierten Amiga 2000-Besitzer beschäftigt, ist der Preis des Turbo-Systems. Eines läßt sich gleich sagen: billig ist es nicht. Zur Grundausstattung gehört auf jeden Fall die Turbo-Karte. Sie schlägt bei einem mit 14 MHz getakteten Arithmetik-Coprozessor mit 3099 Mark zu Buche, die Version mit 25 MHz kostet 4875 Mark. Der Turbo ist aber nur sinnvoll zu nutzen, wenn auch das geeignete RAM zur Verfügung steht. Die 512-KByte-RAM-Karte (statisch) mit 32-Bit-Bus ist für 2895 Mark zu haben. Wem dies nicht genügt, der kann sich auch eine statische 2 MByte-Erweiterung einbauen. Der Preis dieser RAM-Karte: 6495 Mark. Derzeit bietet GIT bei mehr Speicherbedarf zwei dynamische RAM-Karten an: 4 MByte sind für 3995 Mark zu haben, 8 MByte kosten den Interessenten 6495 Mark.

Damit der Turbo-Amiga auch größere Aufgaben im Verbund mit anderen Computern bewältigen kann, bietet GIT die AmeriStar-Netzwerkkarte mit zugehöriger Software zum Preis von 2595 Mark an. Mit dieser Karte lassen sich mehrere Computer vernetzen, so daß daraus eine höhere Rechenleistung und -kapazität resultieren kann.

Ein Thema, das sicherlich viele Interessenten aufhorchen lassen wird: laut Angaben des Vertreibers GIT wird derzeit daran gearbeitet, das verbreitete Betriebssystem »Unix« an den Turbo-Amiga anzupassen. Wie weit die Erfolge bereits vorangeschritten sind, war leider nicht in Erfahrung zu bringen. Wie weiterhin von GIT zu erfahren war, ist derzeit eine Schnittstellenkarte in Vorbereitung, auf der vier serielle Ports enthalten sein sollen. Ebenfalls in Arbeit ist eine bidirektional arbeitende IEEE-488-Karte, mit deren Hilfe sich beispielsweise Steuerungsaufgaben lösen lassen sollen. Sobald diese Karten fertiggestellt sind, werden wir natürlich ausführlich darüber berichten.

Wer sollte sich einen Turbo kaufen? Sicher lohnt sich die Investition nur, wenn der Computer auch kommerziell eingesetzt wird. Überall, wo es auf hohe Rechenleistung ankommt, kann der Turbo wesentlich teurere Computer ersetzen. Denkbare Anwendungen wären schnelle Grafikprogramme, Echtzeitsimulationen oder komplizierte Prozeß-Steuerungen.

Der Turbo bringt auf jeden Fall Bewegung in den Markt. Im Workstationbereich ist er eine ernstzunehmende Konkurrenz. Wenn die Software weiter entwickelt wird und fehlerfrei läuft, stehen dem Turbo alle Türen offen. Sollten die 1-MBit-Chips und die Prozessoren billiger werden, wird der Amiga möglicherweise schon bald standardmäßig mit einem 68020 bestückt. Vielleicht bringt die Zukunft auch noch einen Amiga mit dem noch schnelleren 68030-Prozessor?

Der Turbo in der Industrie

Mitten in unseren Test platzte die Nachricht, daß das Unternehmen MBB-ERNO in Bremen den Turbo zur Simulation der Startphase einer Ariane-Rakete verwendet hat. Das war doch was. Der Amiga im All. Dieser Nachricht mußten wir auf den Grund gehen. Wie setzt ein Luft- und Raumfahrtunternehmen den Amiga ein? Also auf nach Bremen.

Die Zentrale des Luftfahrtunternehmens zu finden war einfach. Direkt neben dem Flugplatz liegt die riesige Anlage. Auch das Gebäude in der Hünefelderstraße 1 war schnell

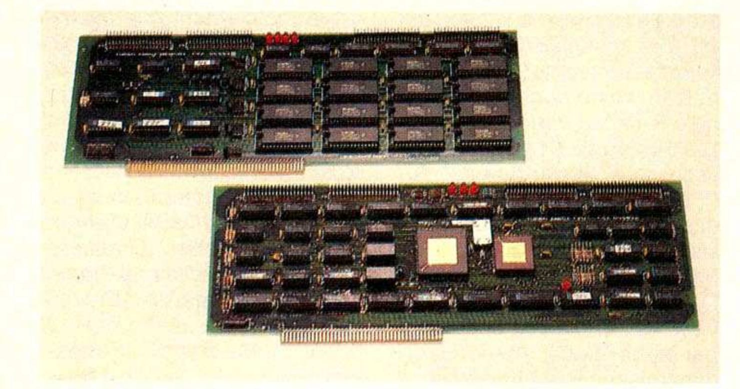


Bild. Die Turbo-Karte mit der RAM-Erweiterung

	Amiga	Turbo	Turbo +Fast-RAM	Turbo +68881	Turbo +68881 +Fast-RAM
Sieb	1	2,7	3,5	2,3	4,1
sin	1	1,8	3,7	17,0	17,8
log	1	1,6	2,9	19,5	20,5
exp	1	1,7	3,1	14,3	17,0
sqrt	1	2,1	4,0	14,1	16,3
arctan	1	1,7	3,4	13,4	18,3
mult	1	1,6	3,1	5,9	7,5
pow	1	1,4	1,6	1,9	2,1
Mand50	1			7,6	7,8
Mand400	1			8,7	9,2
skulpt	1			4,6	6,4

Tabelle. Beschleunigung relativ zum Amiga 2000

gefunden. Hier hat MBB-ERNO seinen Sitz. Das erste, was dem Besucher ins Auge fällt, ist das Schild: »Keine Kameras«. Das war zu erwarten.

Zuerst einmal mußten wir den zuständigen Wissenschaftler informieren, um überhaupt Zutritt zu bekommen. Thomas Görlach betreut das Projekt mit dem Turbo-Amiga. Als Raumfahrttechniker beschäftigt er sich mit der Berechnung und Simulation von Satelliten-Flugbahnen. Da er selbst einen Amiga besitzt und kennt, will er diesen auch bei MBB-ERNO nutzen: »Wenn schon der Amiga, dann muß es ein Turbo sein.« Der Hauptgrund ist die hohe Rechengeschwindigkeit des Computers.

Auch der günstige Preis im Vergleich zu einer 50000 Mark teuren Workstation spielt eine Rolle. Wichtig ist auch die Tatsache, daß die Programmierer nicht nur am Turbo-Amiga arbeiten müssen. Teile der Software lassen sich auf einem ganz normalen Amiga schreiben. Doch noch befindet sich der Einsatz in der Erprobung. Die Kinderkrankheiten des Amiga müssen beseitigt werden. Der zu laute Lüfter wurde beispielsweise ersetzt. Auch das flackernde Bild im Interlace störte. Mit Hilfe eines 4000 Mark teuren Monitors wurde es beseitigt. Was Thomas Görlach bisher mit dem Amiga machte, war die Simulation der Startphase einer Ariane-Rakete zu Demonstrationszwecken. Der gesamte Startflug wurde auf dem Bildschirm dargestellt (Bild 3) und auf Video überspielt:

Das Diagramm zeigt die Höhe der Rakete als Funktion der Zeit. Oben rechts wird die jeweils gezündete Stufe angezeigt.

Das weitere Ziel der ERNO-Techniker ist, mit dem Amiga die bisher eingesetzten VAX 750 und VAX 785 zu entlasten. Gerade wenn es um reine mathematische Berechnungen geht, ist der Turbo schneller als die genannten Rechner. Zusätzlich sollen bei MBB-ERNO eine AT- und eine Netzwerkkarte angeschafft werden. Mit dieser soll der Turbo mit einer SUN-Workstation zusammenarbeiten. Die Netzwerkkarte wurde uns in voller Aktion vorgeführt. Ein Turbo-Amiga mit PC-Karte, 155-MByte-Hard-Disk und 25 MHz getakteten Floatingpoint-Prozessor diente als Server. Ein weiterer Turbo-Amiga wurde angeschlossen und konnte ohne weiteres auf die Hard-Disk zugreifen. Doch noch sind das ungelegte Eier. Erst müssen die anderen Wissenschaftler von der Leistung des Amiga überzeugt werden. Die Software hat noch zu viele Bugs (Fehler). Das kann sich ein Unternehmen wie MBB-ERNO nicht erlauben.

Auf die Frage, in welcher Programmiersprache auf dem Amiga gearbeitet wird, erhielten wir eine spontane Antwort: »Fortran — Fortran ist die Sprache der Wissenschaftler.«

Für alle wichtigen Anwendungen stehen in dieser Sprache bereits Programme zur Verfügung. Der AC-Fortran 77/020-Compiler von Absoft ist nach Meinung von Thomas Görlach ein hervorragendes Instrument, um auf dem Turbo-Amiga Software zu entwickeln. Aber er setzt auch bereits Aztec-C ein. Weniger begeistert ist er von der sonstigen Software: »Ein gutes Textverarbeitungsprogramm fehlt, das auch wissenschaftliche Zeichen verwaltet. Vizawrite war eine herbe Enttäuschung. Wenn die Software erstmal da ist, wird der Turbo bei ERNO noch eine große Rolle spielen.« Dabei wünschen wir T. Görlach und dem Amiga viel Erfolg. (ub)

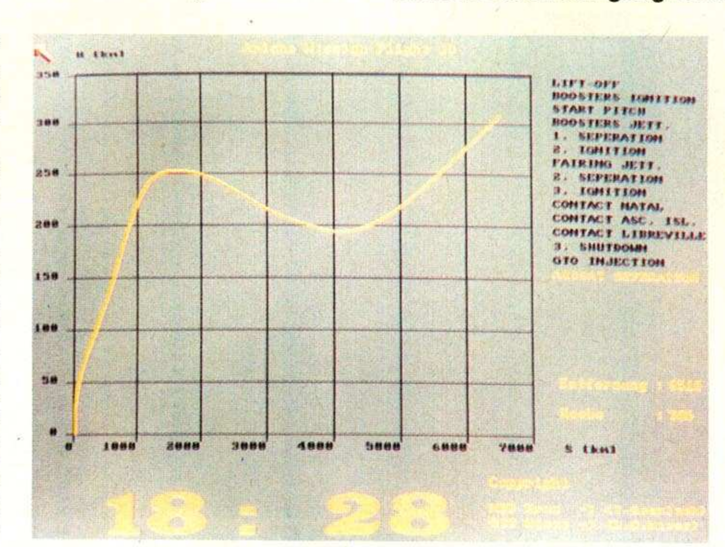


Bild 3. Der Turbo-Amiga simuliert bei MBB-ERNO in Bremen die komplizierte Startphase einer Ariane-Rakete

Motorola heute

Motorola befindet sich im Aufwind. Die Prozessoren der 680xx-Serie sind einer der Gründe. Im Herzen des Amiga schlägt der 68000er — der Vater der Familie. Die Leistung des 68000 wird vom 68020 wesentlich übertroffen. Er besitzt einen größeren Adreßbereich von 4 Gigabyte und einen 32 Bit breiten Datenbus. Die schnellste Version kann mit 20 Megahertz getaktet werden. Eine Neuerung ist der 256 Byte große »Cache-Speicher«. In diesen rettet der Prozessor die zuletzt abgearbeiteten Befehle. Bei einem erneuten Zugriff auf diese Befehle zum Beispiel in einer Schleife braucht der 68020 gar nicht mehr auf den Speicher zugreifen. Was dem 68020 fehlt, ist eine Logik zur Speicherverwaltung. Er benötigt eine externe »Memory-Management-Unit«. Eine solche ist im 68030 bereits integriert. Er benötigt nicht die aufwendige Peripherie seines Vorgängers. Deshalb scheint dieser Super-Prozessor, laut Aussage von Ray Burgess, dem Marketing Direktor von Motorola Europa, noch geeigneter für den Einbau in den Amiga von morgen. Der 68030 übersteigt die Fähigkeiten des 68020 um den Faktor 2. Der »Cache« ist bereits mit 512 Byte ausgelegt. Seine hohe Leistung verdankt der 68030 vor allem der neu eingeführten »Harvard-Architektur«: Unter anderem enthält der Chip zwei interne Daten- und Adreßbusse. Viele Prozesse werden parallel verarbeitet. In Verbindung mit dem ebenfalls neuen Gleitkomma-Coprozessor MC68882 wird der 68030 bestimmt noch für Furore im Personal-Computermarkt sorgen — ganz zu schweigen von dem bereits in der Entwicklung befindlichen 68040, dessen Erscheinen bereits für Ende 1988 geplant ist. (ub)

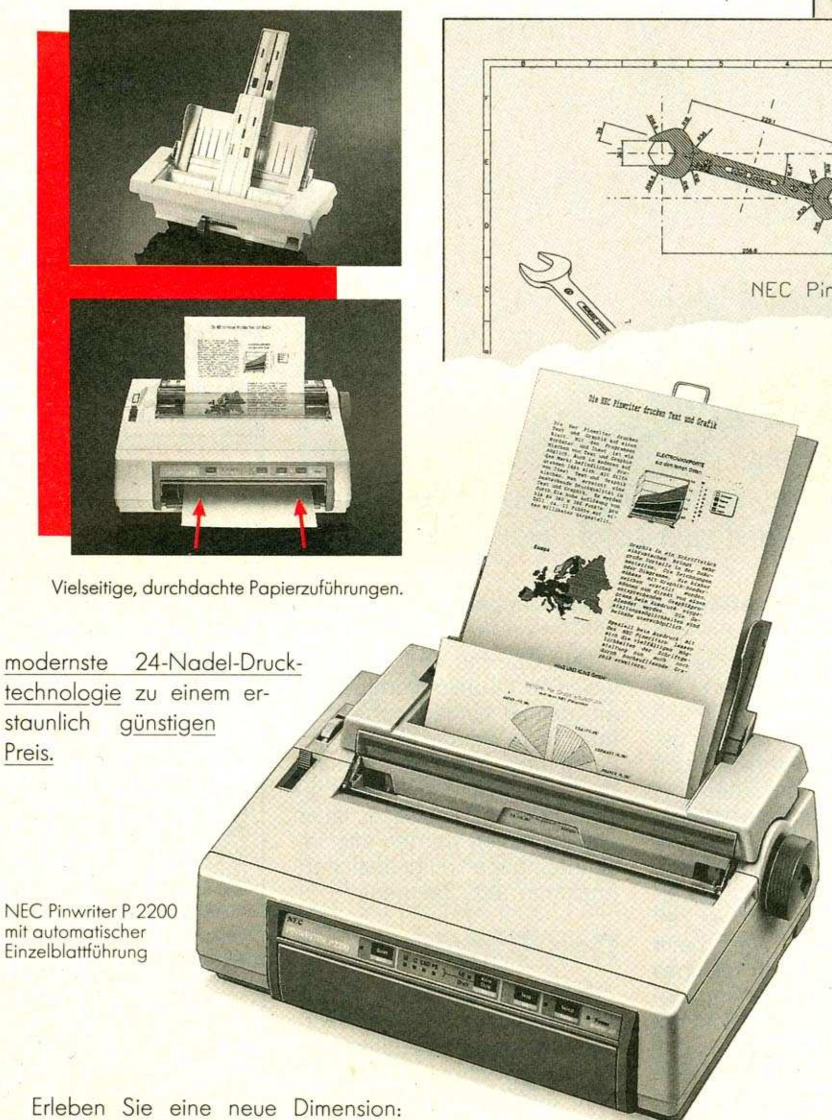
GIT, Gesellschaft für innovative Technologien, Maassenstr. 10, 4235 Schermbeck, Tel. 02853/4099

C&C
Computers and Communications

P 2200 – DAS PREIS-LEISTUNGS-GENIE

Die Computer-Anwender haben Grund zum Jubeln!

Genial – endlich ein Drucker, der für Einsteiger, Aufsteiger und Semiprofis geeignet und vor allem erschwinglich ist. Denn NEC erschließt Ihnen jetzt die modernste 24-Nadel-Drucktechnologie zu einem erstaunlich günstigen Preis.



Vielseitige, durchdachte Papierzuführungen.

Hervorragende Druckqualität durch bewährte 24-Nadel-Technologie.

NEC Pinwriter P 2200 mit automatischer Einzelblattführung

Erleben Sie eine neue Dimension: gestochen scharfen Korrespondenzdruck mit ungewöhnlich reicher Schriftartenauswahl, brillante Grafik-Darstellung, bequeme Druckersteuerung und integrierte Papierzuführungen.

Warum also tief in die Taschen greifen, wenn es schon für wenig Geld 24-Nadel-Technologie mit allen Raffinessen gibt?

PROFIQUALITÄT ZUM AMATEURPREIS

NEC ist mit seinen 24-Nadel-Druckern in Deutschland marktführend.

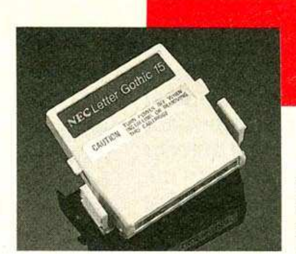
Was den P 2200 als echten Profi auszeichnet, sind seine hohe Auflösung von 360 x 360 dpi, ein halbes Dutzend serienmäßiger Schriftarten und eine Reihe praktischer Papierzuführungen. Zum Beispiel können Sie zwischendurch einen Brief drucken, ohne daß das Endlospapier extra herausgenommen werden muß.

Durch seine volle Kompatibilität mit den NEC Pinwritern der 24-Nadel-Serie harmoniert der P 2200 mit allen wichtigen Software-Paketen.

EIN NEC DRUCKER FÜR JEDERMANN

Endlich braucht niemand mehr auf die bewährte NEC Produkt- und Druckqualität zu verzichten.

Eine breitgefächerte Gruppe – vom Schüler über den Heimanwender bis hin zum Freiberufler – findet im P 2200 die ideale Drucklösung. Anwendern, die schon seit langem auf der Suche nach einem preisgünstigen Drucker für ihren Computer sind, eröffnet er die Möglichkeit, Druckergebnisse in bewährter NEC-Qualität zu erzielen.



12 Schriftartenkassetten zusätzlich erhältlich.

Damit ist der P 2200 die optimale wirtschaftliche und technische Alternative für alle, die sich bei gleichem finanziellen Aufwand bisher nur mit antiquierten 9-Nadel-Druckern begnügen mußten.

Weitere Informationen zum P 2200 erhalten Sie von Ihrem NEC Drucker-Fachhändler.

NEC

NEC Deutschland GmbH

Klausenburger Straße 4, 8000 München 80
Tel.: 0 89/9 30 06-0, Telefax: 0 89/93 77 76/8
Telex: 5 218 073 und 5 218 074 nec m

Eine Uhr mit Pfiff

AMIGA test

Nicht nur für den A500 und A2000 existieren batteriegepufferte Echtzeituhren. Mit dem TimeSaver können nun auch A1000-Besitzer von einer Echtzeituhr mit zusätzlichen, nützlichen Funktionen profitieren.

Wer für seinen Amiga 1000 eine Echtzeituhr sucht, die automatisch beim Starten die Amiga-Uhr stellt, kann mit dem »TimeSaver« von C Ltd. ein Produkt erwerben, das erheblich mehr kann, als nur die Uhrzeit und das Datum zur Verfügung zu stellen. Der Käufer erhält in einem kleinen Plastikkästchen einen 8-Bit-Mikroprozessor, der mit Hilfe eines 8-KByte-EPROMs und 8 KByte RAM eine Reihe vorprogrammierter oder vom Anwender selbst definierter Makros (Befehle) den Amiga steuern kann.

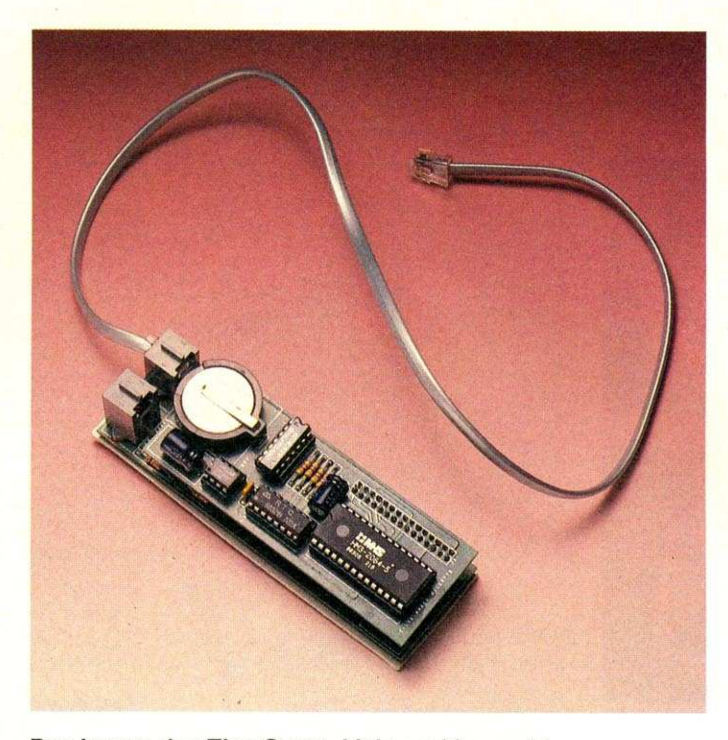
Die Montage des TimeSaver ist problemlos. Das mitgelieferte Kabel wird anstelle des Tastaturanschlusses am Amiga 1000 eingesteckt. Das Tastaturkabel findet seinerseits Anschluß am TimeSaver. Das einzige, was hierbei beachtet werden muß, ist, ob man eine amerikanische oder eine deutsche Tastatur besitzt, da dementsprechend ein amerikanischer oder deutscher TimeSaver gekauft werden muß.

Die hervorstechendste Eigenschaft des Gerätes ist die eingebaute, batteriegepufferte Echtzeituhr, die unabhängig vom Amiga die Uhrzeit und das Datum speichert. Um die eingebaute Batterie zu schonen, wird bei laufendem Computer auf die Amiga-Spannungsversorgung umgeschaltet, mit der die Tastatur betrieben wird.

Die weiteren, in den TimeSaver eingebauten Funktionen lassen sich durch Drücken von <HELP> zusammen mit einer der Funktionstasten aktivieren. Beispielsweise erfährt man das Echtzeitdatum durch die Tastenkombination <HELP> <F5>. Gibt der Anwender <HELP> <F4> ein, läßt sich die Uhr stellen.

Arbeiten mit Makros

Der TimeSaver gestattet das Erstellen von Makros. Makros sind Befehlskombinationen oder ganze Vorgänge, die mit einem einzigen Tastendruck aktiviert werden können. Dies reicht vom einfachen Programmaufruf auf Tastendruck bis hin zu größeren Kommandofolgen, die in etwa mit einer Batch-Datei vergleichbar sind. Eines der so erstellten Makros kann sogar zum »Auto-Makro« ernannt werden. Dieses spezielle Makro wird dann bei jedem Booten ausgeführt. Im Betriebssystem des Geräts sind bereits einige Makros gespeichert, die natürlich auch verändert werden können.



Das Innere des TimeSaver: klein und kompakt

Nützlich ist auch die optionale Paßwort-Funktion. Mit ihrer Hilfe kann der Amiga vor fremden Fingern »geschützt« werden. Das Paßwort darf vier Zeichen umfassen. Wird beim Start des Amiga ein falsches oder gar kein Paßwort eingegeben, kann der Amiga nicht benutzt werden, es sei denn, daß der TimeSaver komplett entfernt wird. Damit gehen aber auch alle im TimeSaver gespeicherten Daten verloren.

Laut Hersteller ist das TimeSaver-Betriebssystem-ROM bisher erst zur Hälfte gefüllt. Weitere Funktionen sollen im Laufe der Zeit implementiert und so das Gerät verbessert werden. Dazu soll, sobald vorhanden, möglicherweise als Update, ein neues ROM erhältlich sein, welches der Benutzer selbst einbauen kann, da alle Chips gesockelt sind.

Das CLI des Amiga ist teilweise etwas unkomfortabel, da ein einmal eingegebener Befehl nicht korrigiert werden kann, falls man sich vertippt hat. Hier hilft der Wiederholspeicher von TimeSaver. Ist er eingeschaltet, werden alle Tastatureingaben mitgespeichert (maximal 1000 Zeichen).

Der Editor

Will man nun einen fehlerhaft eingegebenen Befehl korrigieren, kann der Editor des TimeSaver eingeschaltet werden, der daraufhin die letzte eingegebene Zeile mit einem vorangestellten »;« auf dem Bildschirm ausgibt. Die Zeile läßt sich nun korrigieren, wobei die Cursortasten benutzt werden können. Nach der Korrektur läßt sich der Befehl noch einmal mit <RETURN> starten. Da das vorangestellte Semikolon verhindert, daß das CLI die Änderungen akzeptiert, muß der TimeSaver die korrigierte Zeile neu ausgeben, diesmal ohne das Semikolon, aber mit einem Return am Ende.

Will man einen früheren Befehl ausführen, lassen sich im Editiermodus die einzelnen Befehlszeilen mit den Cursortasten durchblättern, eine durchaus hilfreiche Funktion.

Nicht im Wiederholspeicher werden Repeat-Zeichen (Zeichen, die durch längeres Drücken der Taste wiederholt werden) festgehalten, da diese der Amiga ausführt und nicht von der Tastatur kommen. Daher wird ein solches Zeichen auch nur einmal festgehalten.

Obwohl sich jeder TimeSaver nur an einen bestimmten Tastaturtyp (deutsch oder amerikanisch) anschließen läßt, ist das Gerät sicher allein schon der Makrofunktion wegen eine nützliche Hilfe für A1000-Besitzer. (H. G. Ewald/dm)

AMIGA-WERTUNG

Hardware: TimeSaver

8,8 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■	■	■		
Bedienung	■	■	■	■	■	
Verarbeitung	■	■	■	■	■	
Leistung	■	■	■	■	■	

Fazit: Der TimeSaver ist sicher für Amiga 1000-Besitzer eine überlegenswerte Anschaffung. Allein schon die Echtzeituhr wertet den Amiga auf. Auch der Paßwortschutz und die Makrodefinition tragen zum Komfort des Geräts bei.

Positiv: deutsches Handbuch; belegt keinen Speicher; batteriegepufferte Echtzeituhr eingebaut; Makrodefinition; Paßwortschutz; viele Funktionen; leicht zu bedienen; laut Herstellerangaben Verbesserungen in Vorbereitung.

Negativ: Für deutsche und amerikanische Tastaturen verschiedene Versionen nötig; TimeSaver-Betriebssystem noch nicht in der endgültigen Form.

DATEN

Produkt: TimeSaver
Preis: etwa 140 Mark
Hersteller: C-Ltd.
Anbieter: CompuStore, Fritz-Reuter-Str. 6, 6000 Frankfurt/Main 1, Tel. 069/567399
PDC, Louisenstr. 115, Passage Alter Bahnhof, 6380 Bad Homburg, Tel. 06172/24748



Drei Drucker spielen groß auf

AMIGA test

Drei 24-Nadel-Drucker treten an, um die Herrschaft in der Klasse bis 1200 Mark zu erobern: Der Seikosha SL-80AI, der NEC Pinwriter P2200 und der Epson LQ-500. Kann sich das Triumvirat durchsetzen und wer wird der Kaiser?

Die 24-Nadler kommen. Drei dieser Drucker in der mittleren Preisklasse haben wir getestet. Eines der Ergebnisse des Tests lautet: Die Tage der 9-Nadel-Drucker sind gezählt. Im unteren Preissektor haben sie noch gute Chancen, aber ab 800 Mark wird die Luft dünner. Hier beginnt bereits das Herrschaftsgebiet der 24-Nadel-Drucker. Sie hämmern Texte und Grafiken einfach in einer besseren Auflösung und zudem noch schneller aufs Papier. Allein dieser Faktor rechtfertigt die Anschaffung. Durch die Zeitersparnis beim Drucken — Zeit ist Geld — amortisieren sich die mittlerweile geringen Mehrkosten beim Kauf recht bald. Konsequenterweise stellten sich viele der Druckerhersteller auf den Trend ein. Epson, NEC und Seikosha führen in ihrem Programm mit dem LQ-500, dem P2200 und dem SL-80AI drei der zur Zeit preiswertesten Schwarzweiß-Drucker mit 24 Nadeln (siehe Bild). Auch wenn die offiziellen Listenpreise (siehe Tabelle) teilweise über 1000 Mark liegen, werden Sie bei einem genauen Preisvergleich feststellen, daß einige Händler die Drucker günstiger anbieten. Schauen wir uns gemeinsam an, was das Trio am Amiga leistet.

Jeder der Testkandidaten besitzt eine parallele Schnittstelle zum Anschluß an den Amiga. Die Drucker sind sowohl kompatibel zum LQ-1500 als auch zum IBM-Zeichensatz. Zwei Druckertreiber kommen für den Amiga in Frage:»cbm_mps2xxx« und »Epson_LQ-800«. Der Anschluß an den Amiga gestaltet sich in jedem Fall einfach. Doch damit sind die Gemeinsamkeiten des Trios noch nicht ausgeschöpft: Selbstverständlich arbeiten alle mit Einzel- und Endlospapier. Sie besitzen einen halbautomatischen Papiereinzug und können nachträglich mit einem automatischen Einzelblatteinzug versehen werden.

Schwarzweiß ist angesagt

Die eingebaute Hexdump-Funktion, der Druckerselbsttest und der Papierende-Sensor gehören heute sowieso zum Standard. Jedes Modell beherrscht mehrere Schriftarten, unter anderem mindestens eine in Korrespondenzqualität. Hier machen sich die 24 Nadeln bemerkbar. Die Drucker liefern alle ein ausgezeichnetes Schrift- und Grafikbild in den unterschiedlichsten Auflösungen. Wer eigene Zeichensätze programmiert, kann diese bei jedem der Drucker in einem Teil des bereitgestellten Zeichenpuffers von mindestens 8 KByte speichern. Doch hier treten bereits die ersten Unterschiede auf, der SL-80AI kann satte 16 KByte puffern. Weitere Unterschiede existieren unter anderem in folgenden Punkten:
- Bedienungskomfort
- Zahl der Schriftarten
- grafische Auflösung
- Geschwindigkeit

SL-80 in Vorhand

Der SL-80AI ist der Senator unserer Testkandidaten. Schon vor etwa einem Jahr kam er auf den Markt. Als erster 24-Nadler mit einem Preis unter 1000 Mark sorgte er für Aufsehen. Er spielte bei seinem Erscheinen seine 9-Nadel-Konkurrenten an die Wand.

Beim Aufstellen des Druckers fällt zunächst das Gewicht auf. Der SL-80AI ist recht solide gebaut. Nur der Traktor erweckt wenig Vertrauen.

Vor der Inbetriebnahme muß wie bei jedem Drucker die Grundeinstellung nach dem Einschalten definiert werden. Hierzu befinden sich an der Rückseite zwei gut zugängliche Reihen von DIP-Schaltern. Die Schalterpositionen sind im mitgelieferten guten Handbuch dokumentiert. Ebenfalls zum Lieferumfang gehört der Zugtraktor und die Führungen für Einzelblätter. Die Bedienung des halbautomatischen Einzelblatteinzugs erfolgt über einen Zusatzhebel, der direkt am Drehrad der Walze angebracht ist. Dieser Einzug ist nicht nur praktisch, sondern auch zuverlässig.

Nicht ganz so einwandfrei ist der Traktor. Er läßt sich zwar gut montieren, ist aber etwas wackelig. Das Einfädeln des Papiers fällt leicht, doch die Stachelwalzen lassen sich auf der Kunststoff-Führung nicht genügend fixieren. Sobald das ausgestoßene Endlospapier nicht korrekt hinter den Drucker fällt, kann sich die Lage der Walzen verschieben. Dies führte bei unseren Tests einmal zu Schwierigkeiten. Wer die Gefahr kennt und das Papier immer richtig positioniert, vermeidet jeden Ärger.

Die Steuerung des Druckers erfolgt über das Bedienfeld. Hier finden Sie vier Drucktasten. Mit der »H.Mode«-Taste können Sie zwischen der softwaremäßig eingestellten Betriebsart und dem Hochgeschwindigkeits-Modus umschalten. In diesem gibt der

SL-80AI alle Zeichen in Entwurfsqualität aus. Er akzeptiert dann keine Umschaltsequenzen in den LQ-Modus. Die zusätzliche Funktion zum Einstellen des Randes mit der Tastatur ist außergewöhnlich, allerdings nicht unbedingt erforderlich. Selbstverständlich sind dagegen die Tasten für den Zeilen- und Blattvorschub. Sofort fällt der langsame Zeilenvorschub auf. Er macht sich auch beim Drucken bemerkbar. In der Zeile bewegt sich der SL-80AI noch recht schnell, aber bei jeder Walzenbewegung verliert er an Boden. Der großzügige Puffer von 16 KByte mildert diesen Zeitverlust beim Arbeiten. Wer allerdings mit selbstdefinierten Zeichen arbeiten möchte, muß sich beim Drucken mit 2 KByte Zwischenspeicher begnügen.

Was der SL-80AI ausdruckt, kann sich sehen lassen. Das ausgezeichnete Schriftbild (Bild 1) sucht verständlicherweise in dieser Klasse bei den 9-Nadlern seinesgleichen. Nahezu vollkommen und in einem satten Schwarz präsentiert sich die NLQ-Schrift. Für den Grafikausdruck am Amiga ist der Drucker (siehe Bild 2) ebenfalls geeignet. Jedes der Testbilder druckten wir mit Hilfe von »Deluxe-Paint II«.

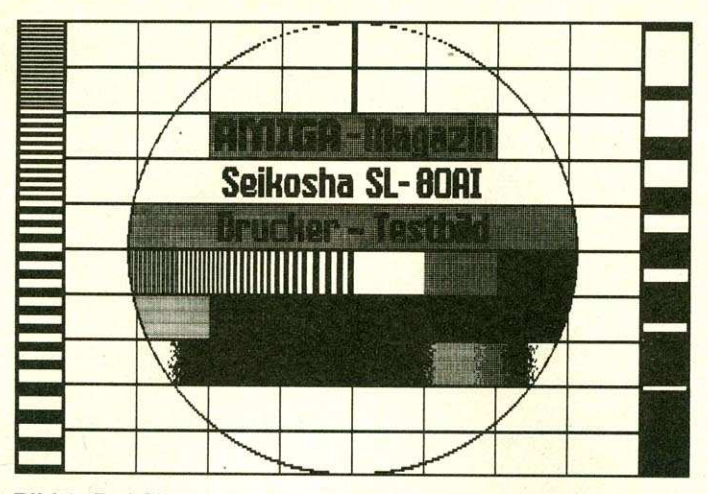


Bild 2. Der SL-80AI druckt 24-Nadel-Grafik mit einer maximalen Auflösung von 180 x 180 Punkten

Seikosha SL-80AI

Normalschrift
Kursivschrift
Fettdruck
Breit
Doppeldruck
Elite Schmalschrift
Hoch und tief
NLQ-Schrift
Kursivschrift
Fettdruck
Breit
Doppeldruck
Hoch und tief

Bild 1. Der SL-80AI druckt auch Schrift sehr satt



AMIGA-WERTUNG

Hardware: Seikosha SL-80AI

8,8 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■	■	■	
Bedienung	■	■	■	■	■	
Verarbeitung	■	■	■	■		
Leistung	■	■	■	■		

Fazit: Der SL-80AI ist ein grundsolider Drucker. Er bietet eine LQ-Schrift in guter Qualität und druckt saubere Amiga-Grafiken. In seiner Leistung vor allem der Druckqualität schlägt er bis 1000 Mark alle 9-Nadel-Drucker.

Positiv: Druckqualität; praktischer und zuverlässiger Papiereinzug; günstiger Preis.

Negativ: Nur ein Zugtraktor; Traktor nicht zuverlässig genug; Papiertransport zu langsam.

DATEN

Produkt: Seikosha SL-80AI

Preis: 899 Mark

Hersteller: Seikosha GmbH, Brahmfelder Chaussee 105, 2000 Hamburg 71

Der im Preference-Menü eingestellte Treshold betrug 10. Dies entspricht einer Auflösung von 180 x 180 Punkten je Zoll. Das ist auch die höchste Auflösung, die der SL-80AI im Grafikdruck mit 24 Nadeln unterstützt. Er bringt dann 1440 Punkte je Zeile aufs Papier. Möchten Sie das Maximum von 1920 Punkten je Zeile ausschöpfen, funktioniert dies mit Treshold 6. Jetzt arbeitet der Drucker im 8-Nadel-Modus und plottet pro Zoll 240 Punkte in der Horizontalen, aber nur 60 in der Vertikalen. Daß die Auflösung noch nicht das Nonplusultra ist, zeigen der LQ-500 und der P2200.

Pinwriters zweiter Stich

NEC stellte kurz vor der Systems den Pinwriter P2200 vor. Der jüngste Sproß dieser Familie beherrscht nicht nur die Befehle des Epson LQ-1500 und den IBM-Zeichensatz, sondern ist auch kompatibel zum P5, P6 und P7. Lediglich der Einbau eines Farbkits ist nicht vorgesehen. Schade eigentlich, aber NEC möchte sich nicht selbst Konkurrenz machen. NEC hat mit dem CP6 und CP7 bereits zwei gute Streiter im Markt positioniert. Der P2200 soll ganz gezielt gegen den IBM-Grafikdrucker, den Epson LQ-800, den Star NB24 und letztlich auch gegen den Seikosha SL-80AI antreten.

Hauptaugenmerk des Druckers ist der offene Schacht in der Front. Er dient zum Einlegen von Einzelblättern. NEC wählte diesen Weg, um die gleichzeitige Nutzung von Einzel- und Endlosblättern zu realisieren. Diesen Service bieten nicht viele Drucker dieser Preiskategorie. Eine Papierparkfunktion gehört bei hochwertigen Bürodruckern mittlerweile zum Standard, beim P2200 ist sie um so erfreulicher. Dieser Komfort erfordert im übrigen einen Schubtraktor. Im Gegensatz zum P6, der ohne Traktor geliefert wird, ist das Teil beim P2200 gleich fest eingebaut. Es handelt sich sogar um einen umschaltbaren Zug- und Schubtraktor — Druckerherz, was willst du mehr. Wer einen Brief schreiben möchte, braucht nun das Endlospapier nicht mehr ein- und auszuspannen. Da stört es den Bediener wenig, daß er zum Einlegen von Endlospapier viel Fingerspitzengefühl benötigt. Es reicht nun, im Offline-Modus die Quiet-Taste zusammen mit der Feed-Taste zu drücken. Der Drucker transportiert auf Ihren Befehl das Papier in eine Warteposition und Sie können mit Einzelblättern arbeiten.

Drücken und schieben

Auch die übrigen Bedienelemente konnten uns gefallen. Alle Schriftarten können mit einer der vier Folien-Tasten angewählt werden. Die Auswahl wird durch kleine Leuchtdioden angezeigt — einfach und praktisch.

Die schon erwähnte Quiet-Taste dient hauptsächlich dazu, die Druckgeschwindigkeit zu verlangsamen und das etwas laute und hochfrequente Druckgeräusch zu dämpfen. Die hohe Tonfrequenz resultiert nicht nur aus der lockeren Bauweise des Druckers, sie läßt sich auch durch das Fehlen der Druckwalze erklären. Der Pinwriter druckt auf einer schmalen Leiste aus Kunststoff, die in einer Metallschiene integriert ist. Vier kleine Gummiwalzen bewegen jedes Blatt. Diese Lösung ist preiswert, birgt aber einige Mängel. Der Transport von Einzelblättern ist nicht zuverlässig genug. Ein Blatt, das mehrmals auf und ab bewegt wird, verschiebt sich häufig. Auch der Einzug von Einzelblättern durch den Frontschacht funktioniert nicht einwandfrei.

Schlecht gezogen

Gespart wurde noch an mehreren Stellen: Vor allem das Gehäuse, der Drehknopf und der Traktor machen einen wackeligen Eindruck. Wie beim Seikosha SL-80AI lassen sich die Stacheln des Traktors nur unzureichend fixieren. Auch die mit dem Druckkopf transportierte Farbbandkasette ist etwas schmächtig. Sie läßt sich allerdings leicht austauschen. Dies gilt auch für den Druckkopf. Er kann vom Bediener leicht gewechselt werden.

Der Verzicht auf DIP-Schalter dagegen ist sinnvoll. Die Wahl der Betriebsart erfolgt über die Tastatur in einem dreistufigen Programmiermodus für folgende Einstellungen:

1. Druckermodus
2. Druckformat
3. Schnittstelle

Die jeweiligen Parameter werden auf dem Papier angezeigt, nach Ihren Wünschen verändert und schließlich in einem CMOS-RAM gespeichert. Im guten Handbuch finden Sie alle Erläuterungen. So dient das erste Menü auch zur Auswahl der nach dem Start verwendeten Schönschrift.

Vier verschiedene Typen (Bild 3) sind im ROM gespeichert. Noch nicht genug? Wer mehr möchte, kann sich einer der zwölf optionalen Font-Kassetten bedienen. In der Schriftwahl liegt eine der Stärken des Druckers. Nicht nur die unterschiedlichen Fonts bestechen, auch die Darstellungsarten überzeugen. Neu sind beispielsweise die Zeichen in dreifacher Breite und in doppelter Höhe. Auch der Grafikausdruck (Bild 4) ist gut. In der höchsten Auflösung plottet der P2200 mit 360 x 360 Punkten pro Zoll. Die Zahl der Punkte je Zeile beträgt 2880.

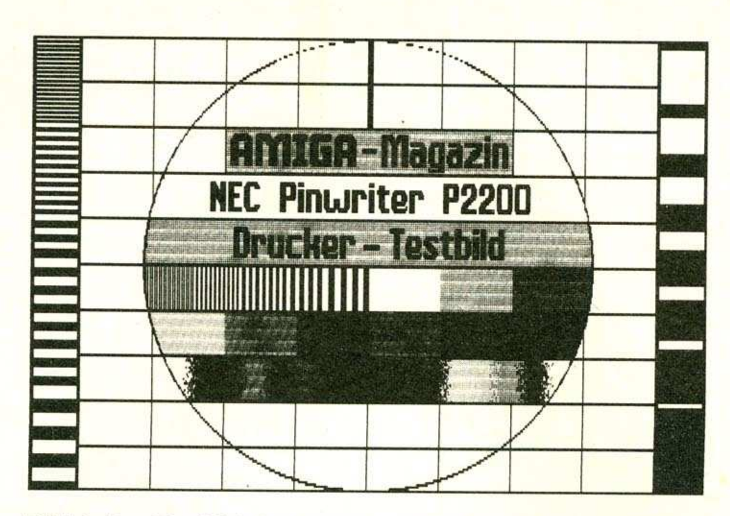


Bild 4. Das Testbild des NEC Pinwriter P2200 wurde mit dem Treshold 10 und Deluxe Paint II gedruckt

Pinwriter P2200
Normalschrift
Kursivschrift
Fettdruck
Breit
Doppeldruck
Elite Schmalschrift
Hoch und tief
NLQ-Schrift
ITC Souvenir 10
Super Focus 10
OCR-B
Courier
Kursivschrift
Fettdruck

Bild 3. Der NEC P2200 bietet insgesamt vier QL-Schriften

Lorbeeren

Ein kompletter Bildschirmausdruck in dieser Auflösung erfordert Geduld. Hier behindert die langsame Steuerung des Amiga. Das Printer-Device und die Druckertreiber sind zu langsam. Commodore und die Druckerhersteller können sich bei der Beseitigung dieses Mankos noch manche Lorbeeren verdienen. Mit über einer halben Stunde dauert der Druck des Testbilds selbst mit dem Treshold 10 zu lang.

Ruhm verdient sich der Pinwriter, wenn es um die Druckgeschwindigeit von Zeichen geht. In der »High-Speed«-Version erreicht er in der Schriftart Elite 142 Zeichen pro Sekunde (angegeben 168). Für unseren Vergleich messen wir die Geschwindigkeit in der Schriftart Pica. Auch hier schneidet der NEC-Drucker gut ab. Im Schönschrift-Modus schafft er immerhin 42 Zeichen pro Sekunde. Noch etwas schneller ist allerdings der Epson LQ-500.

AMIGA-WERTUNG

Hardware: NEC Pinwriter P2200



9,6 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■	■	■	■
Bedienung	■	■	■	■	■	
Verarbeitung	■	■	■	■		
Leistung	■	■	■	■	■	

Fazit: Der P2200 ist ein guter Drucker für den Amiga. Er druckt Texte in ausreichender Geschwindigkeit und guter Qualität. Wer einen Drucker mit extremer Grafikauflösung und reicher Schriftauswahl sucht, findet keinen besseren in dieser Preisklasse.

Positiv: Der integrierte Zug- und Schubtraktor; die Papierparkfunktion für Endlospapier; die große Auswahl unterschiedlicher Schriftarten, sowie die hohe Auflösung im Grafikdruck.

Negativ: Das Einlegen von Endlospapier ist kompliziert; die Papierführung von Einzelblättern läßt zu Wünschen übrig; er druckt etwas zu laut.

DATEN

Produkt: Pinwriter P2200
Preis: 1138 Mark
Hersteller: NEC, Klausenburger Str. 4 8000 München 80

Jung und flink

Epson ist der jüngste des Triumvirats. Sein ansprechendes Design deutet auf die Zielgruppe hin. Der LQ-500 soll auch im semiprofessionellen Bereich Marktanteile gewinnen. Da es sich bei dem getesteten Modell um ein Vorserienexemplar handelte, fehlte auf dem Drucker das Typenschild. Der LQ-500-Schriftzug ist natürlich auf den mittlerweile für 1098 Mark erhältlichen Verkaufsexemplaren angebracht.

Professionellen Anforderungen kommt der Drucker unter anderem durch eine hohe Druckgeschwindigkeit und eine annehmbare Lautstärke entgegen. In unserem Test war er der schnellste. Sowohl der Druckkopf als auch die Walze bewegen sich recht flott. Daß die Geschwindigkeit der Qualität des Ausdrucks keinen Abbruch tut, sehen Sie am Schriftbild (Bild 5).

Die Vorbilder des LQ-500 sind auf der einen Seite der LX-800, auf der anderen der LQ-800. Vom LX-800 stammt das Design und die Bedienung. Der halbautomatische Einzug arbeitet besonders zuverlässig.

Die Wahl der beiden LQ-Schriften »Sans Serif« und »Roman« mit einer Taste ist einfach. Wie beim NEC-Drucker können zusätzliche Schriftmodule nachgerüstet werden. Diese werden in einen gut zugänglichen Schacht im hinteren rechten Drittel des Druckers installiert. In diesem Schacht finden Sie zusätzlich die DIP-Schalter zum Einstellen des Druckers. Die gewählte Konfiguration kann zwecks Überprüfung beim Einschalten ausgedruckt werden.

Weiteres Vorbild des Druckers waren der LQ-800 und der LQ-1500. Zu beiden ist der LQ-500 kompatibel. Nur etwas fehlt, was die großen Brüder auszeichnet: Der Drucker arbeitet nicht mit einem Schubtraktor. Ein Zugtraktor für solch einen Drucker — eigentlich schade. Das Teil ist zwar zuverlässig, doch vermißt ein Kenner der LQ-Serie das Papierhandling des LQ-800.

Damit läßt sich leben, denn der LQ-500 druckt gut (Bild 6). Zwar erreicht er nicht die höchste vertikale Auflösung des P2200, doch unterstützt er ebenfalls den Grafikdruck von 2880 Punkten je Zeile. Damit kommt er auf 360 x 180 Pixel je Zoll.

Eine gute Idee

Bei den Schriften hat sich Epson etwas Besonderes einfallen lassen. Das Angebot an Darstellungsarten wird um die Varianten »Shadow« und »Outline« erweitert. Beide können gemeinsam und mit nahezu allen anderen Schriftarten kombiniert werden. Hoffentlich kommen bald Textverarbeitungsprogramme für den Amiga auf den Markt, die diese Option unterstützen.

Gerade für Schreiber, die ihren Texten eine besondere Note geben möchten, sind die Shadow und Outline hervorragend geeignet. Die Schriften sind zumindestens eindrucksvoller als der übliche Fett-, Breit- oder Doppeldruck. Sicher werden andere Druckerhersteller diese Idee aufgreifen. Eine Idee, die sich letztlich erst durch den Einsatz von 24 Nadeln realisieren läßt.

AMIGA-WERTUNG

Hardware: Epson LQ-500

9,8 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■	■	■	
Bedienung	■	■	■	■	■	
Verarbeitung	■	■	■	■	■	
Leistung	■	■	■	■	■	

Fazit: Der LQ-500 macht einen soliden Eindruck. Ob für Büro- oder Heimanwendungen, der Drucker kann überall eingesetzt werden. Die Schriftqualität und das Drucktempo sind überzeugend.

Positiv: Schnelligkeit; zwei neue Schriftvarianten »Shadow« und »Outline«, die zudem noch beliebig kombiniert werden können; gute Papierführung.

Negativ: Nur ein Zugtraktor. Hier vermißt der Anwender das Papierhandling der anderen Drucker der LQ-Serie von Epson.

DATEN

Produkt: Epson LQ-500
Preis: 1098 Mark
Hersteller: Epson Deutschland, Zülpicher Str. 6, 4000 Düsseldorf 11

Epson LQ-500
Normalschrift
Kursivschrift
Fettdruck
B r e i t
Doppeldruck
Elite Schmalschrift
Hoch und tief
NLQ-Schrift
Sans Serif
Roman
Shadow Outline
Shadow+Outline

Bild 5. Outline und Shadow können überzeugen

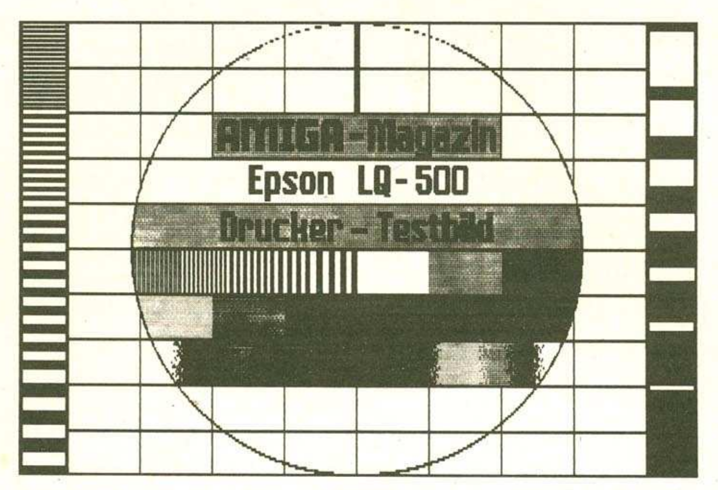


Bild 6. Die mit dem Epson LQ-500 gedruckte Testgrafik offenbart keine Schwächen

Name	Epson LQ-500	NEC P2200	Seikosha SL-80AI
Abmessungen (B x H x T) [mm]	390 x 139 x 320	390 x 140 x 275	419 x 139 x 325
Gewicht [kg]	7,0	5,0	7,3
Tastatur	On Line, FF, LF	On Line, Feed Quiet, Select	On Line, FF, LF H.Mode
Blatt-Transport:			
Halbautomatisch	Standard	Standard	Standard
automatisch	Option	Option	Option
Traktor	Zug aufsetzbar	Zug und Schub eingebaut	Zug aufsetzbar
Druckertreiber	CBM_MPS_2xxx Epson_LQ800	CBM_MPS_2xxx Epson_LQ800	CBM_MPS_2xxx Epson_LQ800
Papierformate [mm]			
Einzelblatt	182 bis 257	165 bis 216	64 bis 254
Endlospapier	101 bis 254	127 bis 254	64 bis 254
Durchschläge	2	2	2
Zeichen/Zeile	max. 160	max. 168	max.160
Schriftarten	Pica, Elite Roman, Sans Serif	Pica, Elite Courier, Letter Gothik Super Focus, ITC Souvenir	Pica, Elite NLQ
Variationen	Breit, Schmal, Italic doppelte Höhe Hoch und Tief Fett und Doppelt Outline, Shadow	Breit, Schmal, Italic doppelte Höhe Hoch und Tief Fett und Doppelt Dreifache Breite	Breit, Schmal, Italic doppelte Höhe Hoch und Tief Fett und Doppelt
Puffer [KByte]	8	8	16
Ladbare Zeichen	128	96	128
Hexdump/Selbsttest	ja/ja	ja/ja	ja/ja
Geschwindigkeiten: [Zeichen/Sekunde]			
EDV PICA 10	130 (150)	109 (140)	90 (108)
NLQ	46 (50)	42 (47)	32 (45)
Testbrief [sek]			
EDV	20,32	26,82	36,12
NLQ	37,65	43,69	56,53
Grafik [min:sek]			
Testbild Treshold 10	27:50	32:30	36:12
Zeichenmatrix			
EDV (H x B)	9 x 23	9 x 17	9 x 17
NLQ (H x B)	29 x 23	32 x 17	29 x 17
Lautstärkeeindruck	mittel	laut	mittel
Grafikmodi:			
9 Nadeln	480,640,720,960,1920	480,640,720,960,1920	480,640,720,960,1620,1920
24 Nadeln	480,720,960,1440,2880	480,720,960,1440,2880	480,720,960,1440
Listenpreise:			
Farbband	28 Mark	17 Mark	24,50 Mark
Drucker	1098 Mark	1138 Mark	899 Mark
Hersteller	Epson Deutschland GmbH, Zülpicher Str. 6, 4000 Düsseldorf 11	NEC Klausenburger Str. 4 8000 München 80	Seikosha GmbH Brahmfelder Chaussee 105, 2000 Hamburg 71

Tabelle. Die wichtigsten Daten der drei Testkandidaten auf einen Blick

24 Nadeln gewinnen

Der Test hat es gezeigt: In ihrer Klasse machen die drei Drucker zur Zeit das Rennen. Jeder hat seine Existenzberechtigung. Der NEC ist bedienerfreundlich, bietet die höchste Auflösung im Grafikdruck, ist aber vor allem äußerlich etwas wackelig. Der Epson ist ein Epson. Das sagt alles. »Flott im Stiling und im Druck«. Es fehlt lediglich das Papierhandling der LQ-Serie beziehungsweise des NEC P2200. Der Seikosha SL-80AI schließlich bietet nicht die Variationen seiner Konkurrenten, ist aber grundsolide. Er hat alles, was ein Drucker haben muß. Schrift und Grafik können sich sehen lassen. Preislich ist er mit einem Listenpreis von 899 Mark etwas günstiger. Was alle drei Drucker für den Amiga vermissen lassen, ist die Option farbig zu drucken. Auch muß sich noch in Sachen Druckertreiber für den Amiga etwas bewegen. Der Grafikausdruck mit 24 Nadeln dauert zu lange. Sicher ist auch interessant zu sehen, was die anderen Druckerhersteller in Zukunft zu bieten haben. Sie werden dem Triumvirat das Feld nicht kampflos überlassen. Wir dürfen gespannt sein. (ub)

Druckertreiber

Wie Sie an dem Druckertest in der AMIGA sehen, stellen wir zu jedem der getesteten Drucker auch die verfügbaren Treiber vor. Jeden Drucker testen wir ausschließlich am Amiga. Somit haben Sie die Gewißheit, wie und ob Sie einen Drucker am Amiga betreiben können. Für zahlreiche Drucker existieren spezielle Treiber, einige andere können mit kompatiblen Treibern betrieben werden. Bei exotischen Modellen ist es eventuell erforderlich einen vorhandenen Treiber etwas zu verändern. Hierzu finden Sie dann in der Rubrik Tips und Tricks Lösungen.

Schnelle Treiber

Einziges Übel mancher Treiber ist zur Zeit die Dauer eines Grafikausdrucks. Hier ist eine dringende Verbesserung des Printer-Device erforderlich. Auf der Workbench 1.3 sollen sich wesentlich verbesserte Druckertreiber befinden.

Immer am Ball mit LigaTab

Viele Amiga-Besitzer sind auch Fußball- oder Handballfans und wollen ständig die Tabelle im Überblick haben. Aber »LigaTab« kann noch mehr. Es verwaltet mehrere Tabellen und liefert viele statistisch relevante Zusatzinformationen.

LigaTab kann zu den vorgegebenen Ligen, von der Bundes- bis zur Kreisliga, auch selbsterstellte Tabellen verwalten, was es besonders flexibel macht. Also finden auch die Freunde von Basketball und anderen Sportarten in LigaTab eine gute Hilfe. Voraussetzung ist allerdings, daß die Punkteverteilung wie beim Fußball stattfindet. Also 2 Punkte für einen Sieg und 1 Punkt bei einem Unentschieden. Außerdem erlaubt das Programm nicht, daß zwei Vereine mehr als zweimal gegeneinander antreten. Bei den zwei Spielen muß jeder Klub einmal daheim und einmal auswärts spielen.



Foto: Jürgen Rossmann

Unter anderem lassen sich auch verschiedene Fußballigen mit »LigaTab« erfassen und verwalten

Das Programm ist bis auf das Eintippen der Vereinsnamen komplett mausgesteuert. Dadurch erhöht sich natürlich die Sicherheit gegenüber Fehleingaben und die Bedienerfreundlichkeit. Außerdem bietet LigaTab noch eine Besonderheit, die später erwähnt wird. Im folgenden wird nun die Bedienung des Programms besprochen, wobei die Menüpunkte der einzelnen Pull-Down-Menüs erklärt werden.

- **PROJECT**

Dieser Menüpunkt beinhaltet die Funktionen, die Sie zur Verwaltung der verschiedenen Ligen benötigen.

— Neue Liga erstellen

Verschiedene Ligen

Wer nicht die vorhandenen Ligen benutzen will, kann hier einen neuen Liganamen sowie alle Klubnamen eintippen. Vor der eigentlichen Eingabe erfolgt noch eine Sicherheitsabfrage, da in diesem Programmteil der gesamte Speicher gelöscht wird und somit auch sämtliche nicht gespeicherte Daten. Es erscheint nun ein eigenes Fenster mit dem Namen »Ligawahl«. 14 verschiedene Ligen sind schon voreingestellt und können durch Anklicken des Namens gewählt werden. Für eine andere Liga geben Sie mit der Tastatur den gewünschten Namen ein. Nach erfolgter Tipparbeit klicken Sie auf das Gadget (Schalter), in dem »OK« steht.

In dem neu aufgebauten Fenster mit dem Namen »Neue Liga« können Sie nun die Namen der Klubs eintragen. Die Eingabe wird immer mit <RETURN> abgeschlossen. Nach dem letzten Verein schließen Sie diesen Programmpunkt ebenfalls durch <RETURN> ab.

— Liga laden

Auch hier erscheint nach der Sicherheitsabfrage wieder das Fenster zur Eingabe des Liganamens. Die Auswahl erfolgt wie oben beschrieben.

— Liga speichern

Hiermit speichern Sie die aktuelle Liga unter dem bekannten Namen.

— Ende

Dieser Menüpunkt dient zum Beenden von LigaTab. Wenn an der Datei Veränderungen vorgenommen wurden, erscheint eine Sicherheitsabfrage. Dort können Sie entscheiden, ob doch noch gespeichert werden soll.

- **FUNKTIONEN**

In diesem Menü finden Sie die Menüpunkte, mit denen Sie sich die verschiedenen Ergebnisse auflisten lassen können.

— Tabelle anzeigen

Wenn Sie neue Daten eingeben, wird die Tabelle auf dem Bildschirm nicht aktualisiert. Wählen Sie diesen Menüpunkt, erscheint die veränderte Liste mit den neuen Informationen. Außer dem Torverhältnis und dem Punktestand gibt LigaTab noch die Anzahl der absolvierten Spiele in der Spalte »SP« aus.

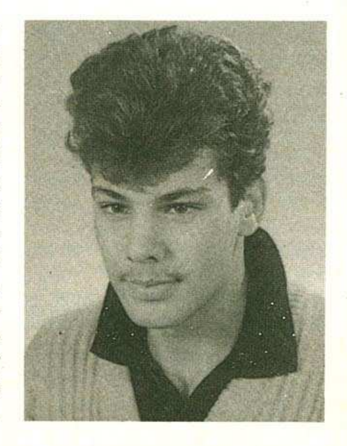
— Einzelbilanz

Auf dem Bildschirm wird ein neues Fenster geöffnet, in dem Sie durch Anklicken des Vereinsnamens den gewünschten Klub auswählen können. Durch Klicken auf den Schalter »Cancel« beenden Sie diesen Programmteil.

Haben Sie jedoch einen Verein gewählt, erscheint ein neues Fenster, in dem eine »Minitabelle« für den entsprechenden Klub aufgebaut wird. Die Bilanz ist nach Heim- und Auswärtsspielen aufgeteilt und zeigt die Siege, Niederlagen, Unentschieden sowie die Punkte und die Tordifferenz an. Im unteren Teil finden Sie dann noch die verbleibenden Spiele und den Tabellenplatz. Wie auch in den folgenden Fenstern gelangen Sie durch Anwahl des Gad-

Jeden Monat 2000 DM GEWINNEN

Daß Torsten Illmann, der Autor des Programm des Monats, ein Hand- und Fußballfan ist, steht fest. Aber auch mit Computern hat er schon länger zu tun. Zwei Jahre lang programmierte er auf dem C 64 in Basic und Assembler. Der Aufstieg zu seinem Amiga 500 erfolgte letztes Jahr im Mai. Außer zur Tabellenverwaltung mit »LigaTab« verwendet er den Amiga auch für Chemieaufgaben und viele Ausdrucke im Gymnasium.



gets »OK« zum nächsten Fenster. Es folgt nun die Auflistung der bisherigen Begegnungen, wobei wieder nach Heim- und Auswärtsspielen unterschieden wird. Im nächsten Fenster zeigt das Programm in der selben Art die noch zu bewältigenden Spiele.

Als Leckerbissen wird dann noch in einer Grafik vorgeführt, wie die Spiele bis dahin gelaufen sind. Dies gibt auf einen Blick eine recht gute Übersicht über die Stärken, Schwächen und die Tendenz einer Mannschaft.

— Gesamtbilanz

Für die gesamte Liga sehen Sie hier die Bilanz für Heim- und Auswärtsspiele und interessante Details wie die Mannschaften mit den meisten und wenigsten Toren, oder der besten und schlechtesten Tordifferenz. Auch die durchschnittliche Toranzahl pro Spiel können Sie erfahren.

- **EDIT**

In diesem Menü finden Sie die Punkte, mit denen Sie einzelne Spiele verändern können.

— Neuer Spieltag

Um die neuen Spielergebnisse einzugeben, wählen Sie durch Anklicken des Namens zweier Vereine zunächst die Gegner aus.

Komfortabel und flexibel

Dabei ist natürlich darauf zu achten, daß die Heimmannschaft als erstes ausgesucht wird. Versuchen Sie nun einen Gegner auszusuchen, gegen den schon gepielt wurde, blinkt der Bildschirm kurz auf und Sie müssen erneut wählen. Die zwei Namen erscheinen dann in den oberen beiden Feldern, wenn nicht eine schon gepielte Partie gewählt wird. Mit den zwei Pfeilen unter dem Ergebnis können Sie nun die Tore erhöhen oder vermindern. Haben Sie für den Heimverein die richtige Zahl erreicht, klicken Sie das OK-Gadget an. Anschließend stellen Sie genauso die Tore für den Gegner ein. Nach erneutem Klicken auf »OK« erscheinen die zwei Schalter »CONTINUE« und »END«. Mit dem ersten Schalter gelangen Sie wieder zur Auswahl der Gegner und können so mehrere Spiele hintereinander eingeben. Wählen Sie jedoch »END«, müssen Sie etwas warten, bis die Ergebnisse in der Liste eingetragen sind. Sie befinden sich dann wieder im Ausgangsfenster und können die Pull-Down-Menüs verwenden.

— Spiel löschen

Es kann trotz allem Komfort natürlich vorkommen, daß Sie ein falsches Ergebnis eingegeben haben. Für diesen Fall gibt es diesen Menüpunkt, mit dem Sie einzelne Spiele löschen können. Zunächst erscheint die Liste aller Vereine, aus der Sie durch Anklicken den Heimverein wählen. Hat der Klub schon ein Heimspiel bestritten, erscheint die Liste aller Vereine die dort schon angetreten sind. Auch hier erfolgt die Auswahl wieder durch einen Klick auf den Vereinsnamen.

Das folgende Fenster ähnelt dem zum Eingeben der Spielresultate. Sie werden hier aber aufgefordert, das falsche Ergebnis einzugeben. Dies ist nötig, da aus Speicherplatzgründen nicht die einzelnen Spielergebnisse gespeichert werden. Sollte dabei ein Fehler entdeckt werden, zeigt LigaTab dies an.

— Teamname ändern

Sollten Sie beim Eintippen der Vereinsnamen Fehler gemacht haben, so können Sie diese leicht ändern. Nach dem Anklicken des zu ändernden Namens können Sie einen neuen Namen eingeben, der dann sofort verwendet wird.

— Liga wechseln

Falls Sie aus Versehen bei der Neuanlage einer Liga den falschen Namen gewählt haben, können Sie hier den Namen ändern. Durch Speichern steht dann die Liga unter dem neuen Namen auf der Diskette.

- **DRUCKEN**

— Tabelle ausdrucken

Natürlich wollen Sie die neue Tabelle nicht nur auf dem Bildschirm sehen. Mit diesem Menüpunkt erhalten Sie die Ergebnisse schwarz auf weiß von Ihrem Drucker. Die Liste sieht auch etwas anders aus als die auf dem Bildschirm. Es werden die Spielanzahl, die Tordifferenz und die Punkte auf das Papier gebracht. Zusätzlich wird die Anzahl der Spiele noch aufgeteilt — je nachdem, ob sie mit einem Sieg, einer Niederlage oder einem Unentschieden endeten.

Bis jetzt waren die meisten Amigas der Sportfreunde am Samstag zwischen 18:05 Uhr und 19:00 Uhr sicherlich ausgeschaltet. Aber das wird sich ja jetzt schnell ändern.

(Torsten Illmann/rb)

Programmname:	LigaTab
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic

```
c23        '      LIGATAB
3L         '   geschrieben von
Es         '   Torsten Illmann
tU0  CLEAR ,62000&
qL   WINDOW 1," *** LigaTab ***   by T. Illmann Juni '87"
N9   PALETTE 0,.1,.27,.13
gD   GOSUB Dimensionierung
pK   Anfang:
pl   GOSUB Tabelle
Fs   ON ERROR GOTO Fault
VN   ON MENU GOSUB Auswahl
Qq   MENU ON
g2   warten:
72   GOTO warten
11   Auswahl:
fc   ON MENU(0) GOSUB PROJECT,FUNKTIONEN,EDITIEREN,DRUCKEN
dF   RETURN
a2   PROJECT:
tJ1   ON MENU(1) GOSUB NeueListe,Laden,Speichern,Ende
gI0  RETURN
L91   NeueListe:
2e2    IF team<>0 THEN GOSUB sicher
q0     IF j=1 THEN RETURN
3r     change=0
iS     ERASE liga$,team$,hwin,olost,owin,hlost,h%,h,o,s,h$,pu,gpu
       ,tore,gtore,hremis,oremis,o$,platz
zW     GOSUB Dimensionierung
x1     GOSUB ligawahl
65     WINDOW 2,"Neue Liga",(10,60)-(400,100),18
F0     team=0
Pz    Eingabe:
393    CLS
q3     team=team+1
RR     LOCATE 2,4
F5     COLOR 1
Af     PRINT "team " USING"##";team;
3b     PRINT" :  ";
R7     LINE (111,7)-(350,17),3,b
MD     COLOR 2
fJ     INPUT "",team$(team)
Sr     IF team$(team)="" THEN weiter1
WR     platz(team)=team
bk2   GOTO Eingabe
91    weiter1:
EJ3    team=team-1
vb2   WINDOW CLOSE 2
6i1  RETURN
f1   Laden:
S42   IF team<>0 THEN GOSUB sicher
GQ    IF j=1 THEN RETURN
K0    GOSUB ligawahl
Lf    OPEN liga$ FOR INPUT AS 1
Te3    INPUT #1,team
ee     FOR i = 1 TO team
jr4     INPUT #1, team$(i),pu(i),gpu(i),tore(i),gtore(i),s(i),hw
```

Listing. Für alle Sportfreunde ist »LigaTab« der ideale Helfer. Bitte mit dem Checksummer (Seite 68) eingeben.

```
          in(i),owin(i)
55 4g     INPUT #1, hremis(i),oremis(i),hlost(i),olost(i),platz(i)
          ,h$(i),o$(i)
56 z43   NEXT
57 jT2  CLOSE 1
58 cY   GOSUB Tabelle
59 cQ   change=0
60 Kw1 RETURN
61 5L  Speichern:
62 d22  IF team=0 THEN BEEP : RETURN
63 xc   OPEN liga$ FOR OUTPUT AS 1
64 Kd3   WRITE #1,team
65 qq    FOR i = 1 TO team
66 aq4    WRITE #1, team$(i),pu(i),gpu(i),tore(i),gtore(i),s(i),hw
          in(i),owin(i)
67 vf     WRITE #1, hremis(i),oremis(i),hlost(i),olost(i),platz(i)
          ,h$(i),o$(i)
68 BG3   NEXT
69 vf2  CLOSE 1
70 nb   change=0
71 V71 RETURN
72 bz  Ende:
73 4R2  IF change=1 THEN GOSUB Saveabfrage
74 Om   MENU RESET
75 lr   CLS
76 yt1  END
77 L60 FUNKTIONEN:
78 it1  ON MENU(1) GOSUB Tabelle,Ebilanz,Gbilanz
79 dFO RETURN
80 lg1  Tabelle:
81 rx2   CLS
82 s6    l=8*(team+3)
83 Gs    LINE (15,6)-(600,l),1,b
84 N3    LINE (15,16)-(600,16)
85 Ie    LINE (295,6)-(295,l)
86 2B    LINE (327,6)-(327,l)
87 48    LINE (426,6)-(426,l)
88 np    LINE (470,6)-(470,l)
89 ro    LINE (560,6)-(560,l)
90 YQ    LOCATE 2,6
91 G8    COLOR 3
92 m6    PRINT  "T E A M N A M E" TAB(39) "SP" TAB(45) "Torverhältn
         is";
93 2t    PRINT TAB(63) "Punktestand"
94 D3    COLOR 1
95 un    IF team = 0 THEN BEEP : RETURN
96 LL    FOR i = 1 TO team
97 WV3    n=0
98 vJ     loop:
99 594     n=n+1
100 u5     IF platz(n)=i THEN weiter2
101 8B3   GOTO loop
102 9m    weiter2:
103 z44    LOCATE 3+i,3
104 mS     PRINT USING"# #";i;
105 kd     PRINT ". " team$(n) TAB(39) USING"# #";s(n);
106 YV     PRINT TAB(44) USING"# # #";tore(n);
107 Oo     LOCATE 3+i,44
108 03     IF tore(n)<100 THEN PRINT "0";
109 jW     IF tore(n)<10  THEN PRINT "0";
110 1l     PRINT TAB(47) ":" USING"# # #";gtore(n);
111 iC     LOCATE 3+i,48
112 ZI     IF gtore(n)<100 THEN PRINT "0";
113 2n     IF gtore(n)<10  THEN PRINT "0";
114 XN     PRINT TAB(55) USING"+# # #";tore(n)-gtore(n);
115 R9     PRINT TAB(63) USING"# #";pu(n);
116 ZO     LOCATE 3+i,63
117 ad     IF pu(n)<10 THEN PRINT "0";
118 Um     PRINT TAB(65) ":" USING"# #";gpu(n);
119 oI     LOCATE 3+i,66
120 80     IF gpu(n)<10 THEN PRINT "0";
121 uQ     PRINT TAB(72) USING"+# #";pu(n)-gpu(n);
122 382   NEXT
123 Lx1  RETURN
124 44   Ebilanz:
125 e32   IF team=0 THEN BEEP : RETURN
126 ao    WINDOW 6,"Einzelbilanz",(200,20)-(550,170),18
127 74    LOCATE 3,5
128 rj    COLOR 3
129 ca    PRINT "Bitte eine Mannschaft wählen :"
130 nd    COLOR 1
131 wx    FOR i=1 TO team STEP 2
132 pJ3    LOCATE 6+(i-1)/2,5
133 tZ     PRINT TAB(5) LEFT$(team$(i),15) TAB(25) LEFT$(team$(i+1),
           15)
134 FK2   NEXT
135 vN    GOSUB maus1
136 iS    WINDOW CLOSE 6
137 t5    IF m=0 THEN RETURN
138 ms    CLS
139 j7    LOCATE 3,33
140 Or    COLOR 2
141 J6    PRINT UCASE$(team$(m))
142 zp    COLOR 1
143 Pb    LINE (244,12)-(268+8*LEN(team$(m)),28),3,b
144 ep    LINE (40,36)-(548,98),1,b
145 nh    LINE (40,52)-(548,52)
146 GO    LINE (178,36)-(178,98)
147 Z7    LINE (208,36)-(208,98)
148 R1    LINE (290,36)-(290,98)
149 bD    LINE (372,36)-(372,98)
150 31    LINE (456,36)-(456,98)
151 hJ    LINE (516,36)-(516,98)
152 KW    LINE (41,83)-(547,83),2
153 G8    COLOR 3
154 6c    LOCATE 6,24
155 Ju    PRINT "SZ" TAB(29) "SIEGE" TAB(39) "ND.LAGEN" TAB(50) "REM
          IS" TAB(59) "PUNKTE" SPC(1) "DIF"
156 60    LOCATE 8,7
157 L1    PRINT "Heimspiele"
158 gc    LOCATE 10,7
159 jv    PRINT "Auswärtsspiele"
160 sq    LOCATE 12,7
161 vD    PRINT "Spiele gesamt"
162 J9    COLOR 1
163 Nv    LOCATE 8,24
164 Fd    PRINT  USING"# #";hwin(m)+hlost(m)+hremis(m);
165 qP    PRINT TAB(31) USING"# #";hwin(m);
166 iE    PRINT TAB(42) USING"# #";hlost(m);
167 X1    PRINT TAB(52) USING"# #";hremis(m);
168 XP    PRINT TAB(60) USING"# #";2*hwin(m)+hremis(m);
169 jx    LOCATE 8,60
170 ms    IF 2*hwin(m)+hremis(m)<10 THEN PRINT "0";
171 Qo    PRINT TAB(62) ":" USING"# #";2*hlost(m)+hremis(m)
172 s9    LOCATE 8,63
173 OO    IF 2*hlost(m)+hremis(m)<10 THEN PRINT "0";
174 cX    PRINT TAB(66) USING"+# #";2*(hwin(m)-hlost(m))
175 lw    LOCATE 10,24
176 nx    PRINT  USING"# #";owin(m)+olost(m)+oremis(m);
177 ip    PRINT TAB(31) USING"# #";owin(m);
178 ae    PRINT TAB(42) USING"# #";olost(m);
179 PR    PRINT TAB(52) USING"# #";oremis(m);
180 kH    PRINT TAB(60) USING"# #";2*owin(m)+oremis(m);
181 ny    LOCATE 10,60
182 15    IF 2*owin(m)+oremis(m)<10 THEN PRINT "0";
183 WZ    PRINT TAB(62) ":" USING"# #";2*olost(m)+oremis(m)
184 zD    LOCATE 10,63
185 U9    IF 2*olost(m)+oremis(m)<10 THEN PRINT "0";
186 ir    PRINT TAB(66) USING"+# #";2*(owin(m)-olost(m))
187 7K    LOCATE 12,24
188 Lo    PRINT  USING"# #";s(m);
189 q9    PRINT TAB(31) USING"# #";hwin(m)+owin(m);
190 To    PRINT TAB(42) USING"# #";hlost(m)+olost(m);
191 tT    PRINT TAB(52) USING"# #";hremis(m)+oremis(m);
192 T7    PRINT TAB(60) USING"# #";pu(m);
193 9M    LOCATE 12,60
194 mo    IF pu(m)<10 THEN PRINT "0";
195 us    PRINT TAB(62) ":" USING"# #";gpu(m)
196 Lb    LOCATE 12,63
197 JA    IF gpu(m)<10 THEN PRINT "0";
198 SY    PRINT TAB(66) USING"+# #";pu(m)-gpu(m)
199 kl    LOCATE 15,7
200 1t    COLOR 3
201 o1    PRINT "TOREANZAHL" TAB(40) "VERBLEIBENDE SPIELE"
202 xn    COLOR 1
203 vK    PRINT TAB(8) "Erzielte Tore  : " USING"# # #";tore(m);
204 ea    PRINT TAB(41) "Restliche Heimspiele     : " USING"# #";(t
          eam-1)-hlost(m)-hwin(m)-hremis(m)
205 3C    PRINT TAB(8) "Gegentore      : " USING"# # #";gtore(m);
206 75    PRINT TAB(41) "Restliche Auswärtsspiele : " USING"# #";(t
```

Listing. Für alle Sportfreunde ist »LigaTab« der ideale Helfer. Bitte mit dem Checksummer (Seite 68) eingeben. (Fortsetzung)

```
        eam-1)-owin(m)-olost(m)-oremis(m)
207 bV  PRINT TAB(8) "Tordifferenz   :" USING"+###";tore(m)-gto
        re(m);
208 P4  PRINT TAB(41) "Restliche Spiele gesamt  : " USING"##";2*
        (team-1)-s(m)
209 A2  COLOR 3
210 5X  LOCATE 20,7 : PRINT "TABELLENPLATZ   :  ";
211 90  COLOR 2
212 NY  PRINT USING"##";platz(m)
213 8y  COLOR 1
214 n5  LOCATE 22,37
215 Sv  LINE (272,164)-(320,180),3,bf
216 dc  PRINT "OK"
217 HH  maus6:
218 kx3  WHILE MOUSE(0) : WEND
219 qi   WHILE MOUSE(0)=0 : WEND
220 4M   IF MOUSE(1)<272 OR MOUSE(1)>320 OR MOUSE(2)<164 OR MOU
        SE(2)>180 THEN maus6
221 7D2 CLS
222 5e  FOR i=1 TO team
223 444   h(i)=0
224 JQ    o(i)=0
225 in2 NEXT
226 eT  h(m)=1:o(m)=1
227 NV  LOCATE 2,2
228 ED  PRINT "Bisherige Spiele :";
229 UM  COLOR 3
230 5r  PRINT TAB(25) "Heimspiele" TAB(52) "Auswärtsspiele"
231 11  LINE (7,17)-(144,17),3,b
232 50  LINE (190,6)-(274,17),2,b
233 5f  LINE (406,6)-(520,17),2,b
234 TJ  COLOR 1
235 xr  IF h$(m)="" THEN
236 Kp4   LOCATE 4,25
237 TN    PRINT "------"
238 mf2 END IF
239 ab  IF o$(m)="" THEN
240 aq4   LOCATE 4,52
241 XR    PRINT "------"
242 qj2 END IF
243 eW  IF h$(m)<>"" THEN
244 PS4   FOR i=1 TO LEN(h$(m)) STEP 2
245 VN6     a=ASC(MID$(h$(m),i,1)):h(a)=1
246 aG      a$=MID$(h$(m),i+1,1)
247 ta      LOCATE 3+(i+1)/2,25
248 Rj      GOSUB auswertung
249 zl      PRINT LEFT$(team$(a),15);
250 ph      COLOR 3
251 LQ      PRINT TAB(41) a$;
252 lb      COLOR 1
253 AF4   NEXT
254 2v2 END IF
255 PO  IF o$(m)<>"" THEN
256 pz4   FOR i=1 TO LEN(o$(m)) STEP 2
257 bh6     a=ASC(MID$(o$(m),i,1)):o(a)=1
258 L8      a$=MID$(o$(m),i+1,1)
259 2j      LOCATE 3+(i+1)/2,52
260 dv      GOSUB auswertung
261 Bx      PRINT LEFT$(team$(a),15);
262 1t      COLOR 3
263 wA      PRINT TAB(68) a$;
264 xn      COLOR 1
265 MR4   NEXT
266 E72 END IF
267 w0  rest:
268 WR  LOCATE 12,4
269 bA  LINE (8,84)-(56,100),3,bf
270 VU  PRINT "OK"
271 11  maus2:
272 cp3  WHILE MOUSE(0) : WEND
273 ia   WHILE MOUSE(0)=0 : WEND
274 dR   IF MOUSE(1)<8 OR MOUSE(1)>56 OR MOUSE(2)<84 OR MOUSE(2
        )>100 THEN maus2
275 z52 CLS
276 AI  LOCATE 2,2
277 Uf  PRINT "Restliche Spiele :";
278 H9  COLOR 3
279 se  PRINT TAB(25) "Heimspiele" TAB(52) "Auswärtsspiele"
280 oo  LINE (7,17)-(144,17),3,b
281 sn  LINE (190,6)-(274,17),2,b
282 sS  LINE (406,6)-(520,17),2,b
283 G6  COLOR 1
284 5I  y=3
285 6f  FOR i=1 TO team
286 1o4   IF h(i)=0 THEN
287 Bb6     y=y+1
288 cE      LOCATE y,25
289 Bs      PRINT team$(i)
290 cV4   END IF
291 mr2 NEXT
292 76  IF y=3 THEN
293 Fk4   LOCATE 4,25
294 OI    PRINT "------"
295 ha2 END IF
296 HU  y=3
297 Ir  FOR i=1 TO team
298 mg4   IF o(i)=0 THEN
299 Nn6     y=y+1
300 ON      LOCATE y,52
301 N4      PRINT team$(i)
302 oh4   END IF
303 y32 NEXT
304 JI  IF y=3 THEN
305 dt4   LOCATE 4,52
306 aU    PRINT "------"
307 tm2 END IF
308 Z1  rest1:
309 B6  LOCATE 12,4
310 Gp  LINE (8,84)-(56,100),3,bf
311 A9  PRINT "OK"
312 ck  maus4:
313 HU3  WHILE MOUSE(0) : WEND
314 NF   WHILE MOUSE(0)=0 : WEND
315 UA   IF MOUSE(1)<8 OR MOUSE(1)>56 OR MOUSE(2)<84 OR MOUSE(2
        )>100 THEN maus4
316 ek2 CLS
317 XN  LOCATE 3,25 : PRINT "Spielbilanz für ";team$(m)
318 1C  LINE (20,100)-(600,100),3
319 LR  LINE (300,25)-(300,175),3
320 G8  LOCATE 12,1
321 NQ  PRINT  "+":PRINT "0":PRINT "-"
322 L7  LOCATE 5,15 : PRINT "zu Hause"TAB(50)"auswärts"
323 4t  xabs=580/(2*team)
324 qZ  yabs=150/(2*team)
325 Ux  PSET (20,100)
326 PS  p=2
327 YO  WHILE MID$(h$(m),p,1)<>""
328 LI3  flag%=0
329 zt   IF MID$(h$(m),p,1)="s" THEN flag%=-1
330 wM   IF MID$(h$(m),p,1)="v" THEN flag%=1
331 CB   LINE -STEP(xabs,yabs*flag%)
332 8C   p=p+2
333 oc2 WEND
334 js  PSET (300,100)
335 Yb  p=2
336 2b  WHILE MID$(o$(m),p,1)<>""
337 UR3  flag%=0
338 hi   IF MID$(o$(m),p,1)="s" THEN flag%=-1
339 eB   IF MID$(o$(m),p,1)="v" THEN flag%=1
340 LK   LINE -STEP(xabs,yabs*flag%)
341 HL   p=p+2
342 x12 WEND
343 tV1 RETURN
344 gi  Gbilanz:
345 Cb2  IF team=0 THEN BEEP : RETURN
346 8E   CLS
347 J1   tore=0
348 6U   LOCATE 3,33
349 Rj   spa$=RIGHT$(liga$,2)
350 BK   IF spa$=".f" OR spa$=".h" THEN
351 Zu4    IF spa$=".f" THEN PRINT "FUSSBALL: ";
352 Dy     IF spa$=".h" THEN PRINT "HANDBALL: ";
353 be     la%=LEN(liga$)+8
354 wf     PRINT UCASE$(LEFT$(liga$,LEN(liga$)-2))
355 K32  ELSE
356 oB4    la%=LEN(liga$)
357 33     PRINT UCASE$(liga$)
358 ib2  END IF
359 wr   LINE (244,12)-(268+8*la%,28),2,b
360 Dh   LINE (40,36)-(580,82),1,b
361 1r   LINE (40,52)-(580,52)
362 SA   LINE (178,36)-(178,82)
363 a4   LINE (216,36)-(216,82)
364 nb   LINE (298,36)-(298,82)
```

```
cA   LINE (380,36)-(380,82)
4i   LINE (464,36)-(464,82)
Mq   LINE (540,36)-(540,82)
jb   COLOR 3
b8   LOCATE 6,25
OU   PRINT "SZ" TAB(30) "SIEGE" TAB(40) "ND.LAGEN" TAB(51) "REM
     IS" TAB(60) "PUNKTE" SPC(3) "DIF."
ZT   LOCATE 8,7
TR   PRINT "Heimbilanz"
95   LOCATE 10,7
33   PRINT "Auswärtsbilanz"
ka   COLOR 1
oM   LOCATE 8,24
AC   hwin=0:hlost=0:hremis=0
bA   FOR i=1 TO team
i84    tore=tore+tore(i)
A0     hwin=hwin+hwin(i)
T4     hlost=hlost+hlost(i)
fs     hremis=hremis+hremis(i)
GL2  NEXT
11   s=hwin+hlost+hremis
LN   PRINT  USING"###";s;
Bw   PRINT TAB(31) USING"###";hwin;
vT   PRINT TAB(42) USING"###";hlost;
Bd   PRINT TAB(52) USING"###";hremis;
V1   PRINT TAB(61) USING"###";2*hwin+hremis;
KZ   LOCATE 8,61
Tv   IF 2*hwin+hremis<100 THEN PRINT "0";
Tf   IF 2*hwin+hremis< 10 THEN PRINT "0";
vj   PRINT TAB(64) ":" USING"###";2*hlost+hremis
Wp   LOCATE 8,65
S3   IF 2*hlost+hremis<100 THEN PRINT "0";
CS   IF 2*hlost+hremis< 10 THEN PRINT "0";
H6   PRINT TAB(69) USING"+###";2*(hwin-hlost)
MX   LOCATE 10,24
Zb   PRINT  USING"###";s;
3Z   PRINT TAB(31) USING"###";hlost;
VI   PRINT TAB(42) USING"###";hwin;
Pr   PRINT TAB(52) USING"###";hremis;
3D   PRINT TAB(61) USING"###";2*hlost+hremis;
Rd   LOCATE 10,61
cD   IF 2*hlost+hremis<100 THEN PRINT "0";
Mc   IF 2*hlost+hremis< 10 THEN PRINT "0";
r4   PRINT TAB(64) ":" USING"###";2*hwin+hremis
hx   LOCATE 10,65
1D   IF 2*hwin+hremis<100 THEN PRINT "0";
1x   IF 2*hwin+hremis< 10 THEN PRINT "0";
uU   PRINT TAB(69) USING"+###";2*(hlost-hwin)
so   LOCATE 12,5
SK   COLOR 3
IO   PRINT "TOREANZAHL"
OE   COLOR 1
cw   tm=1:tw=1:gtm=1:gtw=1:tmdiff=1:twdiff=1
Hr   FOR i=2 TO team
NQ4    IF tore(tm)<tore(i) THEN tm=i
wX     IF tore(tw)>tore(i) THEN tw=i
JG     IF gtore(gtm)<gtore(i) THEN gtm=i
Ib     IF gtore(gtw)>gtore(i) THEN gtw=i
6m     IF tore(tmdiff)-gtore(tmdiff)<tore(i)-gtore(i) THEN tmd
       iff=i
t7     IF tore(twdiff)-gtore(twdiff)>tore(i)-gtore(i) THEN twd
       iff=i
v02  NEXT
bS   COLOR 2
mX   PRINT TAB(6) "Erzielte Tore                 :  " tore;
aQ   COLOR 1
US   PRINT TAB(6) "Torreichste Mannschaft        :  ";team$(tm) TAB
     (56) ": " USING"###";tore(tm);
po   PRINT TAB(6) "Torärmste Mannschaft          :  " team$(tw) TAB
     (56) ": " USING"###";tore(tw);
cm   PRINT TAB(6) "Am meisten Tore bekommen      :  " team$(gtm) TA
     B(56) ": " USING"###";gtore(gtm);
6V   PRINT TAB(6) "Am wenigsten bekommen         :  " team$(gtw) TA
     B(56) ": " USING"###";gtore(gtw);
e9   PRINT TAB(6) "Beste Tordifferenz            :  " team$(tmdiff)
     TAB(56) ":" USING"+###";tore(tmdiff)-gtore(tmdiff);
tF   PRINT TAB(6) "Schlechteste Tordifferenz :  " team$(twdiff)
     TAB(56) ":" USING"+###";tore(twdiff)-gtore(twdiff);
nf   COLOR 3
EA   LOCATE 21,5
IP   PRINT "MANNSCHAFTEN  :  ";
ne   COLOR 2
V7   PRINT team;
sk   COLOR 3
ze   PRINT TAB(30) "TABELLENFÜHRER  :  ";
ri   COLOR 2
dC   FOR i=1 TO team
wQ4    IF platz(i)=1 THEN m=i:i=i+1
FK2  NEXT
xi   PRINT team$(m)
zr   COLOR 3
aY   LOCATE 23,5
SW   PRINT "DURCHSCHNITTLICHE TORANZAHL PRO SPIEL : ";
zq   COLOR 2
Tc   IF s THEN PRINT CINT((tore/s)*2)/2; ELSE PRINT "0";
yo   COLOR 1
eG   RETURN
HNO EDITIEREN:
t61  ON MENU(1) GOSUB Spieltag,Loeschen,Teamname,liga
hJO RETURN
2V1  Spieltag:
6W2   IF team<2 THEN BEEP : RETURN
v6    WINDOW 5,"Neuer Spieltag",(100,0)-(470,185),18
Rr    continue:
45    CLS:m1=0:m2=0
fx    LINE (6,6)-(169,17),3,b
ma    LINE (190,6)-(361,17),3,b
Cj    LINE (159,22)-(201,33),3,b
k0    LINE (180,24)-(180,32),3
tN    LOCATE 2,23:PRINT  "-"
QY    FOR i = 1 TO team STEP 2
E14     LOCATE 6+(i-1)/2,5
an      PRINT LEFT$(team$(i),15) TAB (25) LEFT$(team$(i+1),15)
ej2   NEXT
Km    GOSUB maus1
6u    IF m=0 THEN Aufhoeren
OE    m1=m
mR    LOCATE 2,2:PRINT team$(m1)
h8    Gegner:
Pr    GOSUB maus1
Bz    IF m=0 THEN Aufhoeren
zt    IF m = m1 THEN BEEP : GOTO Gegner
NR    IF INSTR(h$(m1),CHR$(m)) THEN BEEP : GOTO Gegner
AP    m2=m
80    LOCATE 2,25:PRINT team$(m2)
vX    LINE (0,40)-(350,88+8*team),0,bf
xH    GOSUB Ergebnis
29    IF cancel=1 THEN continue
XM    change=1
gd    s(m1)=s(m1)+1:s(m2)=s(m2)+1
ed    tore(m1)=tore(m1)+t(1)
o1    gtore(m1)=gtore(m1)+t(2)
np    tore(m2)=tore(m2)+t(2)
u8    gtore(m2)=gtore(m2)+t(1)
Tz    h$(m1)=h$(m1)+CHR$(m2)
Hx    o$(m2)=o$(m2)+CHR$(m1)
YQ    IF t(1)=t(2) THEN
o14     p1=1:p2=1
ef      hremis(m1)=hremis(m1)+1
Pb      oremis(m2)=oremis(m2)+1
84      h$(m1)=h$(m1)+"u"
y5      o$(m2)=o$(m2)+"u"
dM2   ELSE
jc4     IF t(1)>t(2) THEN
vs6       p1=2:p2=0
7x        hwin(m1)=hwin(m1)+1
YF        olost(m2)=olost(m2)+1
B5        h$(m1)=h$(m1)+"s"
7F        o$(m2)=o$(m2)+"v"
kT4     ELSE
1y6       p1=0:p2=2
CJ        owin(m2)=owin(m2)+1
4V        hlost(m1)=hlost(m1)+1
NK        h$(m1)=h$(m1)+"v"
7C        o$(m2)=o$(m2)+"s"
B44     END IF
C52   END IF
tX    pu(m1)=pu(m1)+p1
57    gpu(m1)=gpu(m1)+p2
61    pu(m2)=pu(m2)+p2
```

Listing. Für alle Sportfreunde ist »LigaTab« der ideale Helfer. Bitte mit dem Checksummer (Seite 68) eingeben. (Fortsetzung)

```
AD   gpu(m2)=gpu(m2)+p1
wA   Aufhoeren:
u0   CLS
W0   LINE (40,40)-(120,64),3,bf
7V   LINE (200,40)-(280,64),3,bf
6j   LOCATE 7,7 : PRINT  "CONTINUE" TAB(29) "END"
6A   maus5:
fs3   WHILE MOUSE(0) : WEND
1d    WHILE MOUSE(0)=0 : WEND
y9    IF MOUSE(2) < 40 OR MOUSE(2) > 64 THEN maus5
3R    IF MOUSE(1)>40  AND MOUSE(1)<120  THEN continue

W6    IF MOUSE(1)>200 AND MOUSE(1)<280 THEN rechnen
2f    GOTO maus5
sJ2  rechnen:
6C   CLS
HN   LOCATE 10,10
fY   PRINT "Bitte Warten ..."
zP   GOSUB Tabsort
3m   WINDOW CLOSE 5
zb1 RETURN
JN  Loeschen:
Ih2  IF team=0 THEN BEEP : RETURN
Us   WINDOW 8,"Spiel löschen!",(10,10)-(400,180),18
PX   LOCATE 2,2
VN   COLOR 3
KD   PRINT "Wählen Sie den Heimverein :"
RH   COLOR 1
ab   FOR i=1 TO team STEP 2
Tx4    LOCATE 6+(i-1)/2,5
p2     PRINT LEFT$(team$(i),15) TAB (25) LEFT$(team$(i+1),15)
ty2  NEXT
Z1   GOSUB maus1
Oe   IF m=0 THEN auf
FT   m1=m
Nw   FOR i=1 TO team
MM4    h(i)=0
z42  NEXT
sr   n=0
PS   FOR i=1 TO LEN(h$(m)) STEP 2
RV4    n=n+1
yk     h(n)=ASC(MID$(h$(m),i,1))
492  NEXT
Fu   IF n=0 THEN auf
Zf   CLS
ph   COLOR 3
1t   LOCATE 2,2
Z9   PRINT team$(m1) " - "
mc   COLOR 1
Hg   FOR i=1 TO n STEP 2
oI4    LOCATE 6+(i-1)/2,5
Pt     PRINT LEFT$(team$(h(i)),15) TAB(25) LEFT$(team$(h(i+1)),
       15)
EJ2  NEXT
ww   maus6:
vN   GOSUB maus1
M0   IF m=0 THEN auf
p0   IF m>n THEN maus6
xz   m2=h(m)
L5   a1=INSTR(h$(m1),CHR$(m2))
Cy   a$=MID$(h$(m1),a1+1,1)
7Q   a2=INSTR(o$(m2),CHR$(m1))
qw   CLS
Xp   LINE (6,6)-(169,17),3,b
eS   LINE (190,6)-(361,17),3,b
4b   LINE (159,22)-(201,33),3,b
cs   LINE (180,24)-(180,32),3
1F   LOCATE 2,23:PRINT  "-"
lo   LOCATE 2,2:PRINT team$(m1) TAB(25) team$(m2)
Cf   LOCATE 8,5:PRINT "Bitte falsches Ergebnis eingeben !"
bv   GOSUB Ergebnis
SR   IF cancel=1 THEN auf
Nb   IF tore(m1)<t(1) OR tore(m2)<t(2) OR gtore(m1)<t(2) OR
     gtore(m2)<t(1) THEN falsch
5x   IF t(1)=t(2) THEN
Rp4    IF a$<>"u" THEN BEEP : GOTO falsch
HK     hremis(m1)=hremis(m1)-1
2G     oremis(m2)=oremis(m2)-1
f8     pu(m1)=pu(m1)-1
Cx     gpu(m1)=gpu(m1)-1
qG     pu(m2)=pu(m2)-1
L8     gpu(m2)=gpu(m2)-1
XQ2  END IF
IB   IF t(1)>t(2) THEN
MQ4    IF a$<>"s" THEN BEEP: GOTO falsch
og     hwin(m1)=hwin(m1)-1
Js     olost(m2)=olost(m2)-1
uJ     pu(m1)=pu(m1)-2
VJ     gpu(m2)=gpu(m2)-2
eX2  END IF
H8   IF t(1)<t(2) THEN
8F4    IF a$<>"v" THEN BEEP :GOTO falsch
p8     hlost(m1)=hlost(m1)-1
vU     owin(m2)=owin(m2)-1
UG     gpu(m1)=gpu(m1)-2
BX     pu(m2)=pu(m2)-2
1e2  END IF
Z0   change=1
no   tore(m1)=tore(m1)-t(1)
16   gtore(m1)=gtore(m1)-t(2)
w0   tore(m2)=tore(m2)-t(2)
7D   gtore(m2)=gtore(m2)-t(1)
Xe   s(m1)=s(m1)-1
jn   s(m2)=s(m2)-1
oW   h$(m1)=LEFT$(h$(m1),a1-1)+RIGHT$(h$(m1),LEN(h$(m1))-a1-1)
Iu   o$(m2)=LEFT$(o$(m2),a2-1)+RIGHT$(o$(m2),LEN(o$(m2))-a2-1)
Xd   CLS
io   LOCATE 10,10
6z   PRINT "Bitte Warten ..."
Qq   GOSUB Tabsort
sq   auf:
kW   WINDOW CLOSE 8
R31 RETURN
AA2  falsch:
f1   CLS
KJ   LOCATE 5,5
Kb   PRINT "Falsche Eingabe !!!"
4N   FOR i=1 TO 2000 : NEXT
a7   GOTO auf
XZ1 Teamname:
sV2  IF team=0 THEN RETURN
Jm   WINDOW 8,"Teamname ändern",(100,20)-(500,180),18
FB   LOCATE 2,5
3v   COLOR 3
An   PRINT "Bitte wählen :
zp   COLOR 1
89   FOR i=1 TO team STEP 2
1V4    LOCATE 6+(i-1)/2,5
Na     PRINT LEFT$(team$(i),15) TAB (25) LEFT$(team$(i+1),15)
RW2  NEXT
7Z   GOSUB maus1
sg   IF m=0 THEN aufh
x3   CLS
7x   COLOR 1
9H   LOCATE 2,2
jY   PRINT "Alter Teamname :  ";team$(m)
JT   LOCATE 4,2
a4   PRINT "Neuer Teamname :  ";
Tr   LINE (150,22)-(400,32),3,b
90   INPUT "",team$(m)
G5   change=1
bd   aufh:
F1   WINDOW CLOSE 8
wY   RETURN
vq1 liga:
Hu3   IF team=0 THEN RETURN
9D    GOSUB ligawahl
NC    change=1
1d    RETURN
1Y0 DRUCKEN:
IU2  don$="[1m"
Qk   dof$="[22m"
6z   IF team = 0 THEN BEEP : RETURN
wE   l$="":IF RIGHT$(liga$,2)=".h" THEN l$="Handball:"+LEFT$(li
     ga$,LEN(liga$)-2):GOTO weitq
5F   IF RIGHT$(liga$,2)=".f" THEN l$="Fussball:"+LEFT$(liga$,LE
     N(liga$)-2) ELSE l$=liga$
5K   weitq:
N1   OPEN "PRT:" FOR OUTPUT AS 1
Mq   PRINT #1,CHR$(27);"c";CHR$(27);"[2w";CHR$(27);"[0z"
hT   PRINT #1,SPACE$(18);CHR$(14);CHR$(27);"[4m";CHR$(27);don$;
     UCASE$(l$);CHR$(27);"[24m";CHR$(27);dof$
mq   PRINT #1,CHR$(27);"D"
II   PRINT #1,CHR$(14);"Mannschaft     SP  G V U  Tordiff.  Punk
     te";CHR$(27);"[4m"
ff   FOR i = 1 TO team
```

```
675 qp3    n=0
676 pM     drloop:
677 PT4     n=n+1
678 eK      IF platz(n)=i THEN drweiter
679 I03    GOTO drloop
680 oF     drweiter:
681 6q4     PRINT #1,USING"##";i;
682 Ds      PRINT #1, ". " team$(n) SPACE$(25-LEN(team$(n))) USING"##   ";s(n);
683 Xy      PRINT #1, USING"  ##";hwin(n)+owin(n);
684 fG      PRINT #1, USING"  ##";hlost(n)+olost(n);
685 14      PRINT #1, USING"  ##      ";hremis(n)+oremis(n);
686 hN      IF tore(n)<10 THEN PRINT #1, "00" USING"#";tore(n);:GOTO weit1
687 eA      IF tore(n)<100 THEN PRINT #1, "0" USING"##";tore(n);:GOTO weit1
688 va      PRINT #1, USING"###";tore(n);
689 Eb      weit1:
690 KO      PRINT #1, ":";
691 CB      IF gtore(n)<10 THEN PRINT #1, "00" USING"#";gtore(n);:GOTO weit2
692 2p      IF gtore(n)<100 THEN PRINT #1, "0" USING"##";gtore(n);:GOTO weit2
693 uU      PRINT #1, USING"###";gtore(n);
694 Pi      weit2:
695 uI      PRINT #1, USING"  +###     ";tore(n)-gtore(n);
696 XB      IF pu(n)<10 THEN PRINT #1, "0" USING"#:";pu(n); ELSE PRINT #1, USING"##:";pu(n);
697 DP      IF gpu(n)<10 THEN PRINT #1, "0" USING"#:";gpu(n); ELSE PRINT #1, USING"##:";gpu(n);
698 tZ      PRINT #1, USING"  +##";pu(n)-gpu(n)
699 MR2   NEXT
700 OS    PRINT #1,CHR$(27);"D";CHR$(27);"[24m"
701 7r    CLOSE 1
702 gI1  RETURN
703 Xe   Ergebnis:
704 L42   t(1)=0:t(2)=0:i=1:cancel=0
705 8Y    LINE (16,64)-(96,88),3,bf
706 Kt    LINE (272,64)-(352,88),3,bf
707 5g    LOCATE 10,7:PRINT "OK"
708 di    LOCATE 10,37:PRINT "CANCEL"
709 LE    LINE (144,38)-(167,49),3,bf
710 B1    LINE (192,38)-(215,49),3,bf
711 D4    COLOR 2
712 k8    LOCATE 6,20:PRINT  "<"TAB(26) ">"
713 ah    schreib:
714 qe    LOCATE 4,21:PRINT  USING"##";t(1);
715 Y5    PRINT  SPC(1) USING"##";t(2)
716 F5    COLOR 1
717 n0    WHILE MOUSE(0) : WEND
718 SO    maus7:
719 oa3    FOR z=1 TO 600 : NEXT
720 rk     dummi=MOUSE(0)
721 wo     WHILE MOUSE(0)=0 : WEND
722 Fc     IF MOUSE (2) >38 AND MOUSE (2) <49 THEN Zaehlung
723 BK     IF MOUSE (2) >64 AND MOUSE (2) <88 THEN Abfrage
724 Ot     GOTO maus7
725 W42   Zaehlung:
726 Kf3    IF MOUSE (1) >144 AND MOUSE(1) <167 THEN
727 Pa5      t(i)=t(i)-1
728 Rv       IF t(i)=-1 THEN t(i)=99
729 ha3    END IF
730 vN     IF MOUSE (1) >192 AND MOUSE(1) <215 THEN
731 Ha5      t(i)=t(i)+1
732 70       IF t(i)=100 THEN t(i)=0
733 le3    END IF
734 Ay     LOCATE 4,21:PRINT  USING"##";t(1);
735 bZ     IF i=2 THEN PRINT SPC(1) USING"##";t(2)
736 a5     GOTO maus7
737 sP2   Abfrage:
738 pA3    IF MOUSE(1) > 16 AND MOUSE(1) < 96 THEN
739 vp4     i=i+1
740 AN      WHILE MOUSE(0) : WEND
741 Y7      IF i=3 THEN fortsetzen
742 3j      GOTO schreib
743 vo3    END IF
744 iz     IF MOUSE(1) > 272 AND MOUSE(1) < 352 THEN cancel=1
745 FO     fortsetzen:
746 0O2   RETURN
747 r91  Tabsort:
748 I12   FOR n=1 TO team-1
749 CG3    s=0
750 qt     FOR i=1 TO team-1
751 Sd5      z=0
752 Rp       loop1:
753 nF       z=z+1
754 6z       IF platz(z)<>i THEN loop1
755 Sb       x=0
756 bv       loop2:
757 f3       x=x+1
758 Q2       IF platz(x)<>i+1 THEN loop2
759 xw       GOSUB label
760 LQ3     NEXT
761 Rj      IF s=0 THEN RETURN
762 NS2   NEXT
763 fH1  RETURN
764 wu   label:
765 df2   IF pu(z) = pu(x) THEN
766 704     IF gpu(z) > gpu(x) THEN tausch
767 28      IF gpu(z) = gpu(x) THEN
768 g46       IF tore(z)-gtore(z)<tore(x)-gtore(x) THEN tausch
769 LA        IF tore(z)-gtore(z)=tore(x)-gtore(x) THEN
770 EF8         IF tore(z)<tore(x) THEN tausch
771 NG6       END IF
772 OH4     END IF
773 PI2   END IF
774 8j    IF pu(z) < pu(x) THEN tausch
775 rT1  RETURN
776 cz   tausch:
777 Vj2   s=s+1
778 12    SWAP platz(z),platz(x)
779 vX1  RETURN
780 sC   maus1:
781 Jf2   LOCATE 17,19
782 pZ    IF m1=0 THEN
783 yD3    LINE (136,123)-(200,140),2,bf
784 a9     PRINT "CANCEL""
785 bU2   END IF
786 u7    WHILE MOUSE(0) : WEND
787 Os    WHILE MOUSE(0)=0 : WEND
788 nH    IF MOUSE(1)>136 AND MOUSE(1)<200 AND MOUSE(2)<140 AND MOUSE(2)>124 THEN
789 ec4     m=0
790 6i      RETURN
791 ha2   END IF
792 Cs    IF MOUSE (2) < 40 OR MOUSE(2) > 44+8*team THEN maus1
793 fJ    IF MOUSE (1) < 40 OR MOUSE(1) > 320 THEN maus1
794 Vc    m=2*INT((MOUSE(2)-32)/8)-1
795 il    IF MOUSE(1)>200 THEN m=m+1
796 jn    IF m>team THEN maus1
797 Dp1  RETURN
798 KB   auswertung:
799 Kt2   IF a$="s" THEN a$="(Sieg)"
800 Yx    IF a$="v" THEN a$="(Nd.lage)
801 QT    IF a$="u" THEN a$="(Remis)"
802 Iu1  RETURN
803 sv   sicher:
804 J22   WINDOW 4,"Sicher ?",(100,50)-(450,100),18
805 N3    LOCATE 2,9
806 kb    COLOR 2
807 HJ    PRINT  "Alle Daten im Speicher werden"
808 YA    LOCATE 4,18
809 IO    PRINT  "gelöscht !!!"
810 1b    COLOR 1
811 hA    LINE (50,20)-(95,37),3,bf
812 G2    LINE (250,20)-(295,37),3,bf
813 UU    LOCATE 4,9: PRINT  "OK"
814 9u    LOCATE 4,34:PRINT  "NO"
815 yt    j=0
816 Uo    maus:
817 UM    WHILE MOUSE(0)=0 : WEND
818 zy    IF MOUSE(2)<20 OR MOUSE(2)>37 THEN maus
819 YR    IF MOUSE(1)>50 AND MOUSE(1)<97 THEN weiter3
820 gE    IF MOUSE(1)<250 OR MOUSE(1)>295 THEN maus
821 84    j=1
822 oS    weiter3:
823 dL    WINDOW CLOSE 4
824 eG    RETURN
825 ub1  Saveabfrage:
826 wN2   WINDOW 9,"Ende",(10,20)-(350,110),0
```

Listing. Für alle Sportfreunde ist »LigaTab« der ideale Helfer. Bitte mit dem Checksummer (Seite 68) eingeben. (Fortsetzung)

```
3B      LOCATE 2,2
Y3      PRINT "Aktuelle Datei '";
7y      COLOR 2
a4      PRINT liga$;
6w      COLOR 1
bX      PRINT "' wurde"
HR      LOCATE 4,2
Gc      PRINT "verändert, aber nicht abgespeichert.
1r      LOCATE 8,4
H9      COLOR 3
t3      PRINT "SAVE ???"
D3      COLOR 1
cg      LOCATE 6,20 : PRINT "YES"
j9      LOCATE 8,20 : PRINT "NO"
ML      LOCATE 10,18: PRINT "CANCEL";
iD      LINE (148,38)-(180,50),1,b
t0      LINE (148,54)-(180,66),1,b
o6      LINE (132,70)-(188,82),1,b
hV      maus9:
s5      WHILE MOUSE(0) : WEND
yq      WHILE MOUSE(0)=0 : WEND
QL      IF MOUSE(1)>148 AND MOUSE(1)<180 AND MOUSE(2)>38 AND MO
        USE(2)<50 THEN
vf4       AREA (148,38) : AREA (148,50) : AREA (180,50) : AREA (18
          0,38)
rr        AREAFILL 1
df        GOSUB Speichern
30        GOTO zurueck
ha2     END IF
uu      IF MOUSE(1)>148 AND MOUSE(1)<180 AND MOUSE(2)>54 AND MO
        USE(2)<66 THEN
es4       AREA (148,54) : AREA (148,66) : AREA (180,66) : AREA (18
          0,54)
xx        AREAFILL 1
8T        GOTO zurueck
mf2     END IF
Mn      IF MOUSE(1)>132 AND MOUSE(1)<188 AND MOUSE(2)>70 AND MO
        USE(2)<82 THEN
S04       AREA (132,70) : AREA (132,82) : AREA (188,82) : AREA (18
          8,70)
22        AREAFILL 1
fS        WINDOW CLOSE 9
26        GOTO Anfang
sl2     END IF
rE      GOTO maus9
H3      zurueck:
kX      WINDOW CLOSE 9
My1    RETURN
pf0  ligawahl:
OY1    WINDOW 7,"Ligawahl",(50,50)-(400,170),18
1t     LOCATE 2,2
I3     PRINT  "Liganame :  "
rT     LINE (103,7)-(300,17),3,b
8A     LOCATE 4,4
um     COLOR 3
OW     PRINT "FUSSBALL:" TAB (20) "HANDBALL:"
qg     COLOR 1
Zx     FOR i=1 TO 7
fn2     LOCATE 4+i,5
05      PRINT liga$(i) TAB(21) liga$(i+7)
IN1    NEXT
oY     LINE (140,92)-(180,108),1,bf
zq     COLOR 2
FP     LOCATE 13,20
QP     PRINT "OK"
AK     liga$="":LOCATE 2,15
gm     ON MOUSE GOSUB maus10
Pg     MOUSE ON
Oy3      a$="":mangar=0
Bh       WHILE a$<>CHR$(13) OR liga$=""
jb5        warteschleife:
U2         a$=INKEY$
tQ         IF mangar THEN RETURN
jw         IF a$="" THEN warteschleife
5m         IF ASC(a$)<>47 AND ASC(a$)>31 AND ASC(a$)<123 THEN
Ag7          liga$=liga$+a$
Pa           PRINT a$;
QJ5        END IF
yJ         IF ASC(a$)=8 AND LEN(liga$)>0 THEN
1F7          liga$=LEFT$(liga$,LEN(liga$)-1)
Te           PRINT a$;
UN5        END IF
0o3        WEND
a0         GOTO mangar
pP1    maus10:
y92     m0=MOUSE(0)
Vj      IF MOUSE(2) > 92 AND MOUSE(2) < 108 THEN
VB3       IF MOUSE(1) < 140 OR MOUSE(1) > 180 THEN RETURN
I0        IF liga$="" THEN RETURN
GV        mangar:
W5        mangar=1
L6        MOUSE OFF
K5        WINDOW CLOSE 7
6i        RETURN
ha2     END IF
jz      IF MOUSE(1) < 32 OR MOUSE(1) > 280 THEN RETURN
MV      IF MOUSE(2) < 32 OR MOUSE(2) > 88 THEN RETURN
1c      l=INT((MOUSE(2)-24)/8)
GP      IF MOUSE(1) > 160 THEN l=l+7
8s      IF l>7 THEN spa$=".h" ELSE spa$=".f"
fQ      liga$=liga$(l)+spa$
bm      LOCATE 2,15:PRINT SPACE$(23)
Bi      LOCATE 2,15
6a      PRINT liga$;
Ht1    RETURN
t90  Fault:
Yw1    IF ERR=53 THEN
9B2     WINDOW 3
HC      LOCATE 5,6
TT      PRINT "Datei nicht vorhanden !!!"
FS      WHILE MOUSE(0) : WEND
g1      WHILE MOUSE(0)=0 AND INKEY$="" : WEND
K1      WINDOW CLOSE 3
Ej      RESUME Anfang
1u1    END IF
ek     CLS
qB     PRINT "Fehler:";ERR
Ki     MENU RESET
DG     ON ERROR GOTO 0
up     END
Ic0  Dimensionierung:
I11    DIM team$(30),pu(30),gpu(30),hwin(30),hlost(30),hremis(30)
H8     DIM platz(30),owin(30),olost(30),oremis(30),tore(30),gtore(
       30),s(30)
04     DIM liga$(30),h$(30),o$(30),h(30),o(30),h%(team)
Bx     liga$(1)="Bundesliga"
C8     liga$(2)="2.Bundesliga"
23     liga$(3)="Oberliga"
jc     liga$(4)="Verbandsliga"
nT     liga$(5)="Landesliga"
J4     liga$(6)="Bezirksliga"
kU     liga$(7)="Kreisliga"
dW     liga$(8)="Bundesliga"
eh     liga$(9)="2.Bundesliga"
64     liga$(10)="Regionalliga"
Pq     liga$(11)="Oberliga"
k4     liga$(12)="Landesliga"
Li     liga$(13)="Bezirksliga"
1H     liga$(14)="Kreisliga"
F0     team=0
g0     MENU 1,0,1," PROJECT  "
S7     MENU 1,1,1,"Neue Liga erstellen"
uY     MENU 1,2,1,"Liga laden          "
05     MENU 1,3,1,"Liga speichern      "
8m     MENU 1,4,1,"Ende                "
pi     MENU 2,0,1,"  FUNKTIONEN  "
uv     MENU 2,1,1,"Tabelle anzeigen"
up     MENU 2,2,1,"Einzelbilanz    "
No     MENU 2,3,1,"Gesamtbilanz    "
aJ     MENU 3,0,1,"  EDIT  "
Pk     MENU 3,1,1,"Neuer Spieltag "
Zi     MENU 3,2,1,"Spiel löschen  "
G1     MENU 3,3,1,"Teamname ändern"
iA     MENU 3,4,1,"Liga wechseln  "
P1     MENU 4,0,1,"    DRUCKEN        "
uY     MENU 4,1,1,"Tabelle ausdrucken
6i     RETURN
(C) 1988 M&T
```

Listing. Für alle Sportfreunde ist »LigaTab« der ideale Helfer. Bitte mit dem Checksummer (Seite 68) eingeben. (Schluß)

Copper-Listen

Viele haben schon vom Copper und den dazugehörigen Listen gehört. Wenn auch Sie daran interessiert sind, solche Listen anzusehen, brauchen Sie »CopperMon«.

Der Copper ist der Spezialchip, der für viele Bereiche der Grafik zuständig ist. Er versetzt zum Beispiel den Amiga in die Lage, mehrere Bildschirme verschiedener Auflösung gleichzeitig darzustellen. Genauso werden die virtuellen Sprites erst durch den Copper möglich. Er ist also einer der wichtigsten Bausteine im ganzen System.

Die Steuerung dieses Spezialchips erfolgt durch nur drei Befehle:

```
WAIT          MOVE          SKIP
```

Das sieht zunächst nach sehr wenig aus, aber die erzielbaren Ergebnisse sind fantastisch. In Ausgabe 11, Seite 84, finden Sie weitere Erklärungen zu diesem Thema. Außerdem können Sie dort sehen, was auch mit Basic erreichbar ist. Das dort abgedruckte Programm können Sie verwenden, um eine Copper-Liste zu erzeugen und dann mit CopperMon (siehe Listing) betrachten.

Nun zu dem Programm selbst. Es ist sehr einfach zu bedienen. Nach dem Start mit dem Befehl

```
coppermon
```

erscheint zunächst eine kleine Einschaltmeldung auf dem Bildschirm, in der auch die Bedienung kurz erklärt wird. Sie können nun durch Drücken der linken Maustaste die Copper-Liste erscheinen lassen. Es werden allerdings nicht die Hexcodes angezeigt, sondern eine Art disassemblierter Code. CopperMon interpretiert nämlich die Steuercodes der Liste und gibt die Information aufbereitet aus.

Bei MOVE-Befehlen steht vorne der Hexwert, der verwendet wird, und hinten die Adresse, in die der Wert geschrieben wird. Besonders gut gelungen ist die Anzeige des WAIT-Befehls. Hinter jedem WAIT steht zunächst die Angabe, bis zu welcher Horizontalposition (»hp«) der Copper wartet. Dahinter kommt dieselbe Angabe für die Vertikalposition (»vp«). Am Schluß steht dann noch der Wert der verwendeten Maske. Bei dem SKIP-Befehl ist leider nur ersichtlich, daß gesprungen werden soll, aber nicht wohin. Dies wäre aber auch sinnlos, da Sie praktisch immer nur einen Teil der gesamten Liste auf dem Bildschirm sehen können.

Nun zur Programmierung von CopperMon. Nach dem Einbinden der nötigen Headerdateien und dem Öffnen der Grafikbibliothek (»graphics.library«) in den Zeilen 2 bis 15 wird es interessant. Der Variablen »adresse« wird in Zeile 16 der Anfang der Copper-Liste zugewiesen. Über mehrere Zeiger erreicht man den Wert »CopLStart«, der auf den Speicherplatz zeigt, wo die Copper-Liste beginnt. Diese Adresse erscheint dann bei der Ausgabe auch als erstes in der Zeile.

Die while-Schleife (Zeilen 22 bis 27) ruft die Funktion »decode« solange auf, wie der linke Mausknopf gedrückt ist. Dabei wird aus der Speicherstelle, auf die »leftmouse« zeigt, der Wert geholt und mit 64 verknüpft.

Die if-Abfrage in den Zeilen 28 bis 32 dient zum Beenden von CopperMon, wenn der Joystick-Knopf gedrückt wird.

Die Funktion decode ist sozusagen das Herzstück des Programms. Es wird die Adresse übergeben, deren Inhalt decodiert werden soll. Enthält Bit 0 des Wertes eine 0 (Zeile 41), so handelt es sich um einen MOVE-Befehl. In der nächsten Zeile wird dann der Text ausgegeben. An erster Stelle steht die Adresse (»copw1«). Danach folgt der Wert, mit dem gefüllt werden soll. Dies ist der Inhalt der auf copw1 folgenden Adresse (»*copw2«). Zu guter Letzt folgt noch die Adresse, in die der Wert geschrieben wird (»*copw1 & 511«).

Handelt es sich um einen WAIT- oder SKIP-Befehl, ist das Bit 0 des Wertes 1 (Zeile 43). Nun muß aufgrund des Bits 0 des zweiten Wertes noch entschieden werden, welcher Befehl vorliegt (Zeilen 45 und 47). In Zeile 46 erfolgt für den WAIT-Befehl die Ausgabe der Zeile. Die Verknüpfungen, um die einzelnen Werte für die horizontale und vertikale Position zu erhalten, ergeben sich aus dem Aufbau des Befehls. Die horizontale Position steht in den Bits 1 bis 7 des ersten Datenworts. Die vertikale befindet sich in den Bits 8 bis 15 und muß daher noch durch 256 geteilt werden. Das zweite Datenwort schließlich enthält die zu verwendende Bitmaske.

Falls es sich um einen SKIP-Befehl handelt, wird in Zeile 47 nur die Adresse und ein Text ausgegeben.

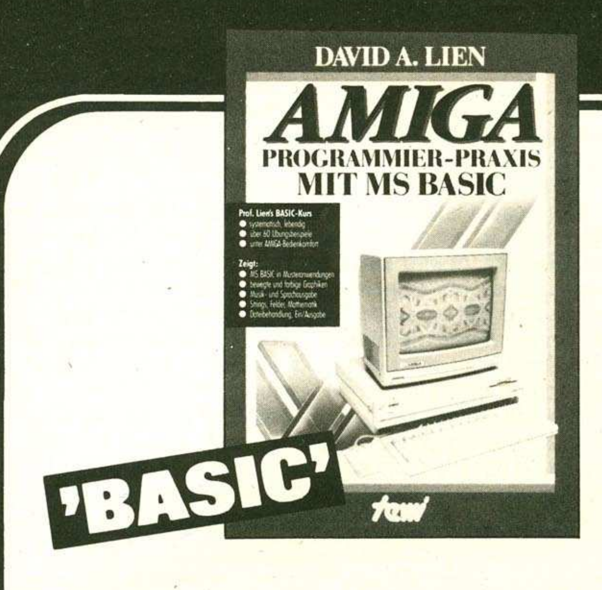
Durch Experimente mit dem CopperMon erfahren Sie viel über die »unsichtbaren« Arbeiten des Amiga. Wenn Sie beispielsweise zwei Bildschirme gleichzeitig auf dem Monitor sehen, wird die Copperliste schon deutlich umfangreicher und komplizierter. Daraus können Sie auch Rückschlüsse ziehen, warum der Amiga langsamer wird, je höher die gewählte Auflösung ist. Aber auch zum Überprüfen selbstgeschriebener Copper-Listen eignet sich CopperMon hervorragend. (Klaus Pfaadt/rb)

Programmname:	CopperMon
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4
Aufrufe:	cc coppermon +l ln coppermon -lc32

```
 1 nV0 /* CopperMon von Klaus Pfaadt */
 2 5z  #include <exec/types.h>
 3 hY  #include <graphics/gfx.h>
 4 bk  #include <graphics/gfxbase.h>
 5 Q4  #include <graphics/view.h>
 6 Bq  #include <graphics/copper.h>
 7 tW  struct GfxBase *GfxBase;
 8 ZD  struct CopList *cp;
 9 w6  UWORD *adresse;
10 su  char *leftmouse;
11 ik  extern APTR OpenLibrary();
12 tF  main()
13 9c  {
14 4n3    leftmouse=0xbfe001;
15 ed     GfxBase=(struct GfxBase*)OpenLibrary("graphics.library",0L
          );
16 dv     adresse=GfxBase->ActiView->ViewPort->DspIns->CopLStart
          ;
17 SB     printf("Coppermonitor aktiviert!\n\n");
18 op     printf(" Listen  - linke Maustaste\n");
19 Ed     printf(" Beenden - Joystick Feuer\n\n\n");
20 Gx     for(;;)
21 Hk     {
22 SX6       while((*leftmouse&0x40)==0)
23 Jm        {
24 KM9          decode(adresse);
25 Mx           adresse+=2;
26 45           WaitTOF();
27 Rw6       }
28 bA        if((*leftmouse&0x80)==0)
29 Ps        {
30 uQ9          CloseLibrary(GfxBase);
31 SR           Exit(TRUE);
32 W16       }
33 X23    }
34 Y30 }
35 tJ  decode(adr)
36 Tu  UWORD *adr;
37 X0  {
38 Fo3    UWORD *copw1,*copw2;
39 8H     copw1=adr;
40 3b     copw2=copw1+1;
41 71     if ((*copw1&1)==0)
42 L66       printf("$%x: move $%x,reg.$%x\n",copw1,*copw2,*copw1 &
             511);
43 C73    if ((*copw1&1)==1)
44 e7     {
45 D86       if ((*copw2&1)==0)
46 o89          printf("$%x: wait hp=%d,vp=%d,mask=$%x\n",copw1,*cop
                w1&254,(*copw1&0xff00)/256,*copw2);
47 x06       if((*copw2&1)==1)printf("$%x: skip !!!???!!!\n",copw1);

48 mH3    }
49 nI0 }
(C) 1988 M&T
```

Listing. »CopperMon« zeigt Copper-Listen im Klartext an. Bitte mit dem Checksummer (Seite 68) eingeben.





NEU

AMIGA: Programmierpraxis mit MS BASIC
David A. Lien
„87% aller PC-Benutzer programmieren in BASIC" (Marktanalyse '87). BASIC ist schnell erlernbar. AMIGA bietet den Programmierkomfort. Hier eine lebendige, systematische Spracheinführung. Sie zeigt: richtige Befehlsanwendung an über 60 Musterprogrammen; bewegte und farbige Graphiken; Musik- und Sprachausgabe; Mathematik und Stringbehandlung; Dateibehandlung; Ein/Ausgabe usw.
426 Seiten. Softcover. DM 59,–

AMIGA: Systemprogrammierung in 'C'
John Th. Berry
Das Buchmotiv: 'C' statt 'Assembler' für den Hi-Tech-PC 'AMIGA'. AmigaDOS, Kernel, Intuition sind eine Schatzkiste an Routinen. Über 100 Beispiele zeigen 'C'-Programmierung mit diesen Routinen. Mit detaillierter Erklärung der Routinen und Übergabeparameter. Behandelt u.a. AMIGA-Hardware, Gadgets, Fenster, Menüs, Dialogboxen, message ports, AmigaDOS-Multiprocessing, Sprites, künstl. Sprache.
464 Seiten. Softcover. DM 59,–

Fordern Sie unseren neuen Commodore-Prospekt an.



Weitere te-wi-Bücher



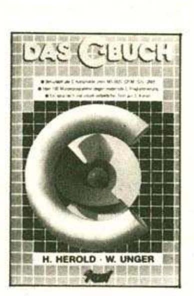
M68000 FAMILIE, 2 Bd.
Hilf/Nausch, ges. 968 Seiten
Einzige Motorola-authentische Darstellung von CPU-68000-Architektur, Programmierung, Systemaufbauten. Behandelt alle 68000-Bausteine sowie 68020, 68881.
Bd 1, Grundlagen + Architektur, 568 Seiten, DM 79,–
Bd 2, Anwendung und Bausteine, 400 Seiten, DM 69,–



dBASE III+ **NEU**
Einführung + Referenz
Update eines Bestsellers! dBASE III+ in 60 Textmodulen: zur Einführung in Reihenfolge eines dBASE-Kurses lesbar, danach als alphabetisches dBASE-Befehlslexikon. Behandelt die Funktions- und Befehlserweiterungen von dBASE III+.
R.A. Stultz. 480 Seiten. Hardcover, DM 79,–



BASIC-Programmierung
GW-BASIC für PC KOMPATIBLE
Eine systematische, kurserprobte (GW)BASIC-Einführung für PC-10/IBM PC-Systeme und Kompatible. Mit Einführung in das System und Tastendarstellung im Text. BASIC-Befehlsbeschreibungen mit Aufgaben und Antworten. Beispielprogramme. Von David Lien. 488 Seiten, Softcover. DM 59,–



DAS C-BUCH. **NEU**
Textbuch für C-Kurse und C-Anwendungen auf PCs. Beschreibt sämtliche Konstrukte der C-Sprache unter den Betriebssystemen MS DOS, CP/M, ISIS, UNIX und für die C-Compiler von MS, DR, LATTICE, INTEL. Didaktisch und typografisch außergewöhnlich.
Mit über 100 lauffähigen Beispielprogrammen für PCs. Zeigt Realisierungen neuester Softwarestrategien in „C".
Von Herold/Unger. 576 Seiten, Softcover, DM 79,–



UMWELTDYNAMIK
30 Programme für kybernetische Umwelterfahrungen auf allen BASIC-Rechnern. Das Buch enthält beides: Ein Programmsystem zur Simulation eigener Problemformulierungen und 29 kommentierte Modellbeispiele wie Baumsterben, Heizungsbedarf, Nahrungsketten usw. Prospekt anfordern.
Von Hartmut Bossel, 480 Seiten, Softcover, DM 59,–



C-64 Computerhandbuch.
Mit GEOS-Update. Raeto West.
C-64 Bestseller! Referenzbuch für Professionelle; Lehrbuch für C-64-Autodidakten; Anwenderhandbuch für C-64-Programmierer. Beantwortet auch seltene, nirgends behandelte Systemfragen. Beste Rezensionen. Mit GEOS-Update!
704 Seiten, Softcover, DM 66,–

704-Seiten-Antwort auf alle C-64-Systemfragen.

tm 4936

Verdeckte Bildschirme – kein Problem

Der Amiga ist ein Multitasking-Computer, was viele Vorteile bringt. Aber was tun, wenn ein im Hintergrund liegender Screen nicht mehr zu erreichen ist? Die Antwort darauf bietet »GetScreen«.

Es ist Ihnen sicher auch schon passiert, daß zum Beispiel ein Basic-Programm mit Grafikbildschirm es unmöglich macht, an einen anderen Screen heranzukommen. Mit <Left Amiga n> kommt man zwar immer auf den Workbenchscreen, aber <Left Amiga m> zeigt immer nur einen der anderen Bildschirme an. Dieses Problem werden Sie nie mehr haben, wenn Sie GetScreen besitzen.

Das Programm (siehe Listing) öffnet nach dem Starten mit

```
run getscreen
```

ein kleines eigenes Fenster. Wie durch den Fenstertitel schon ersichtlich, müssen Sie dieses Fenster anklicken, damit es sich öffnet. In dem Fenster erscheinen nun so viele Zeilen wie verschiedene Bildschirme vorhanden sind. Maximal werden 10 Screens angezeigt, vor denen die Zahlen 0 bis 9 stehen. Der Text dahinter ist der Bildschirmtitel wie er in der Menüzeile des entsprechenden Screens steht.

Um auf den gewünschten Bildschirm zu kommen, klicken Sie einfach auf die Zahl vor dem Text. Schon befinden Sie sich auf dem jeweiligen Screen und können ihn durch Anklicken aktivieren.

Auch wenn Sie auf einem Bildschirm ohne Hintergrundschalter (ToBack-Gadget) sind, kommen Sie immer durch Drücken von <Left Amiga n> auf den Workbench-Screen. Dort sehen Sie wieder das verkleinerte Fenster von GetScreen, das Sie jetzt erneut benutzen können.

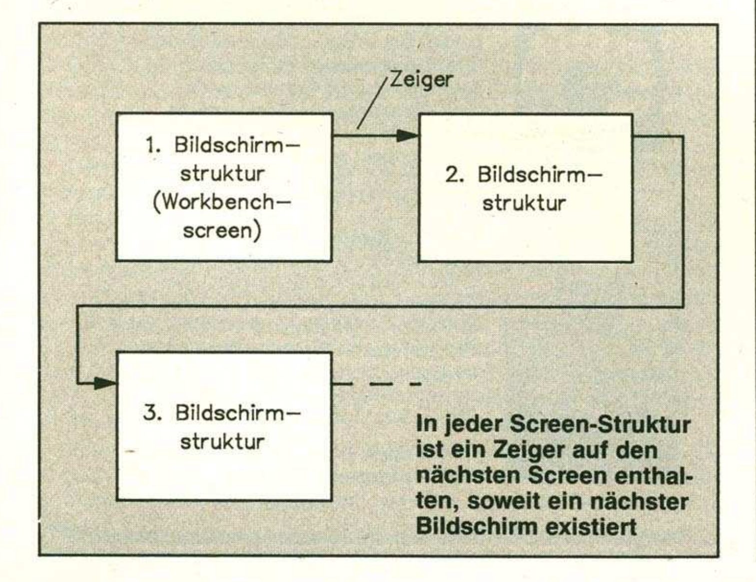
Soweit zu der sehr einfachen Bedienung des Programms. Doch in GetScreen stecken ein paar Funktionen, die höchst interessant sind. Nach den Definitionen (Zeile 5 bis 24), folgt die Funktion zum Öffnen des Fensters. Hier werden die nötigen Bibliotheken (Libraries) eröffnet und die Parameter für das Fenster festgelegt.

Der interessanteste Programmteil ist die Funktion »baue_window_auf«. Zuerst wird die Anzahl der vorhandenen Bildschirme bestimmt (Zeilen 67 bis 73). Dazu muß man wissen, daß in jeder Screen-Struktur ein Zeiger auf den nächsten Bildschirm, soweit vorhanden, steht. Die Definition der Struktur finden Sie im Headerfile »intuition/intuition.h«. Wie Sie dort sehen können, ist der Zeiger gleich das erste Strukturelement.

Interessant ist auch noch die Funktion »hole_screen« (Zeilen 144 bis 153), mit der der ausgewählte Bildschirm in den Vordergrund gebracht wird. Dazu wird zunächst aufgrund der Mausposition ausgerechnet, um welche Bildschirmnummer es sich handelt (Zeile 148). Mit einer »while«-Schleife wird nun solange der nächste Screen ermittelt, bis der Zähler die richtige Nummer erreicht hat. Dann wird mit der Funktion »ScreenToFront« der jeweilige Bildschirm in die vorderste Position gebracht.

Wie Sie sehen, kann man durch den geschickten Einsatz der Betriebssystemfunktionen viel auf dem Amiga erreichen. Es lohnt sich also auf alle Fälle sich Bücher zu besorgen, in denen diese Funktionen beschrieben sind. Denn damit lassen sich sicher noch wahre Wunderdinge auf dem Amiga verwirklichen.

(Gunter Lahm/rb)



In jeder Screen-Struktur ist ein Zeiger auf den nächsten Screen enthalten, soweit ein nächster Bildschirm existiert

Programmname:	GetScreen
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Lattice-C V3.1 oder Aztec-C V3.4
Aufrufe:	siehe Text

```
 1 J00 /************************
 2 Ns  * GetScreen Version 1.1 *
 3 PI  *     Gunter Lahm      *
 4 gv  ************************/
 5 ow  #include "exec/types.h"
 6 c5  #include "intuition/intuition.h"
 7 y0  #include "intuition/intuitionbase.h"
 8 iJ  #define CWIDTH 200
 9 dE  #define AWIDTH 400
10 bJ  #define MAXHEIGHT 256
11 Q6  #define MAXSCREENS 10
12 Uc  #define GS_VERSION "GetScreen v1.1   1987 by  Gunter Lahm"
13 g9  struct IntuitionBase *IntuitionBase;
14 Od  struct GfxBase *GfxBase;
15 5s  struct NewWindow NewWindow;
16 kG  struct Window *GS_Window;
17 q0  struct Gadget GadgetList[MAXSCREENS];
18 cm  struct IntuiText  GText[MAXSCREENS];
19 2M  struct IntuiMessage *Message;
20 XL  ULONG Class;
21 9q  USHORT Code;
22 XN  char Nummer[MAXSCREENS][2];
23 3Q  int wh, anzahl_screens, aufgebaut = FALSE;
24 30  VOID wprint(), ausstieg();
25 rr  VOID fenster()
26 Mp  {
27 Yt2   IntuitionBase = (struct IntuitionBase *)OpenLibrary("intuit
         ion.library",0);
28 JB    if (IntuitionBase == NULL) exit(FALSE);
29 fI    GfxBase = (struct GfxBase *) OpenLibrary ("graphics.library
         ",0);
30 NZ    if (GfxBase == NULL)
31 Ru3    {
32 Zg5      CloseLibrary (IntuitionBase);
33 Ee       exit(FALSE);
34 Y33    }
35 oT2   NewWindow.LeftEdge = 0;
36 3v    NewWindow.TopEdge = 10;
37 3x    NewWindow.Width = CWIDTH;
38 Ab    NewWindow.Height = 25;
39 jD    NewWindow.DetailPen = 0;
40 gW    NewWindow.BlockPen = 1;
41 Xe    NewWindow.Title = GS_VERSION;
42 zH    NewWindow.Flags = SMART_REFRESH | NOCAREREFRESH | WINDOWCLO
         SE |
43 gaK                   WINDOWDEPTH | WINDOWDRAG;
44 Ws2   NewWindow.IDCMPFlags = CLOSEWINDOW | INACTIVEWINDOW | GADGE
         TUP |
45 1qP                        MOUSEBUTTONS;
46 q72   NewWindow.Type = WBENCHSCREEN;
47 vY    NewWindow.FirstGadget = NULL;
48 o8    NewWindow.CheckMark = NULL;
49 72    NewWindow.Screen = NULL;
50 kQ    NewWindow.BitMap = NULL;
```

```
a8   NewWindow.MinWidth = NULL;
LT   NewWindow.MinHeight = NULL;
sS   NewWindow.MaxWidth = NULL;
dn   NewWindow.MaxHeight = NULL;
rx   if ((GS_Window = (struct Window *)OpenWindow(&NewWindow)) =
     = NULL)
qJ3   {
4x5    CloseLibrary (GfxBase);
z6     CloseLibrary (IntuitionBase);
e4     exit(FALSE);
yT3   }
tS2  wprint (GS_Window, "Click here to open", 25, 19);
OVO }
aY  baue_window_auf ()
NF  { int i, dx, dy;
wv2  struct Screen *a_screen;
WX   /* Anzahl der Screens bestimmen */
b7   anzahl_screens = 0;
4b   a_screen = IntuitionBase->ActiveScreen;
TF   while ((a_screen->NextScreen != 0) && (anzahl_screens < M
     AXSCREENS-1))
4X   {
FV3   anzahl_screens++;
tf    a_screen = a_screen->NextScreen;
Bg2  }
pv   wh = 40 + anzahl_screens * 10;
pV   dx = 0;
ta   dy = 0;
ch   if (GS_Window->LeftEdge > 640-AWIDTH) dx = 640-AWIDTH - G
     S_Window->LeftEdge;
vF   if (GS_Window->TopEdge > MAXHEIGHT - wh - 10)
o6M                      dy = MAXHEIGHT - wh - GS_Window->TopEd
                         ge - 10;
232  wprint (GS_Window, "                    ", 25, 19);
zZ   MoveWindow (GS_Window, dx, dy);
XR   SizeWindow (GS_Window, AWIDTH - CWIDTH, wh-15);
LF   while (GS_Window->Width != AWIDTH) Delay(5);
8G   wprint (GS_Window, "Screen- or WindowTitle:", 30, 20);
Lg   i=0;
Mt   a_screen = IntuitionBase->ActiveScreen;
Yi   while (i <= anzahl_screens)
Mp   {
Mr3   if (a_screen->Title != NULL)
F86      wprint (GS_Window, a_screen->Title, 30, 40+i*10);
I53   else
aU4    if (a_screen->DefaultTitle != NULL)
gI7      wprint (GS_Window, a_screen->DefaultTitle, 30, 40+i*1
         0);
L84    else
Yw6      if (a_screen->FirstWindow->Title != NULL)
z29        wprint (GS_Window, a_screen->FirstWindow->Title, 3
           0, 40+i*10);
OB6      else
Ow9        wprint (GS_Window, "No Title", 30, 40+i*10);
bw3   /* Gadgets anbringen */
t8     if (i<anzahl_screens) GadgetList[i].NextGadget = &Gadget
       List[i+1];
KW     else GadgetList[i].NextGadget = NULL;
eI     GadgetList[i].LeftEdge = 11;
16     GadgetList[i].TopEdge = 34+i*10;
sL     GadgetList[i].Width = 10;
rX     GadgetList[i].Height = 8;
Nr     GadgetList[i].Flags = GADGHCOMP;
zn     GadgetList[i].Activation = GADGIMMEDIATE | RELVERIFY;
ze     GadgetList[i].GadgetType = BOOLGADGET;
E1     GadgetList[i].GadgetRender = NULL;
dm     GadgetList[i].SelectRender = NULL;
Go     GadgetList[i].GadgetText = &GText[i];
AG     GadgetList[i].MutualExclude = NULL;
XV     GadgetList[i].SpecialInfo = NULL;
PA     GadgetList[i].GadgetID = i;
ne     GadgetList[i].UserData = NULL;
3Q     Nummer[i][0] = i + 48;
q2     Nummer[i][1] = 0;
em     GText[i].FrontPen = 1;
hD     GText[i].BackPen = 0;
lu     GText[i].DrawMode = JAM1;
Yr     GText[i].LeftEdge = 0;
dr     GText[i].TopEdge = 0;
vO     GText[i].ITextFont = NULL;
oc     GText[i].IText = &Nummer[i][0];
UE     GText[i].NextText = NULL;
zg     AddGadget (GS_Window, &GadgetList[i], -1);
pn     i++;
nZ     a_screen = a_screen->NextScreen;
5a2   }
V5    RefreshGadgets (&GadgetList[0], GS_Window, NULL);
61    return(TRUE);
8d0 }
8J  baue_window_ab ()
6Z  {
fB1  int i;
qt   i = 0;
Mu   while ( i <= anzahl_screens) RemoveGadget (GS_Window, &Gad
     getList[i++]);
re   /* Fenster auf "Wartegroesse" bringen */
J8   wprint (GS_Window, "                          ", 30, 20);
M9   SizeWindow (GS_Window, CWIDTH - AWIDTH, 15-wh);
Bk   wprint (GS_Window, "Click here to open", 25, 19);
Of   return (FALSE);
JoO }
oK  VOID hole_screen ()
Hk  {
6M1  int i, ScreenNr;
GF   struct Screen *a_screen;
K9   ScreenNr = (GS_Window->MouseY - 34) / 10;
36   i = 0;
Ov   a_screen = IntuitionBase->ActiveScreen;
9H   while ((i++ != ScreenNr) && (a_screen)) a_screen = a_screen
     ->NextScreen;
9H   if (a_screen) ScreenToFront (a_screen);
TyO }
bt  VOID wprint (wd, text, x, y)
OW  struct Window *wd;
Sv  char text[];
2D  int x,y;
Ux  {
cb1  Move (wd->RPort, x, y);
Rv   Text (wd->RPort, text, strlen(text));
b60 }
uW  VOID ausstieg ()
Z2  {
H52   ClearMenuStrip (GS_Window);
bx    CloseWindow (GS_Window);
pi    CloseLibrary (GfxBase);
kr    CloseLibrary (IntuitionBase);
81    exit (TRUE);
JEO }
OE  VOID main()
hA  {
Hf1  fenster();
L2   FOREVER
kD   {
Tt3   Wait (1 << GS_Window->UserPort->mp_SigBit);
jV    while (Message = (struct IntuiMessage *) GetMsg (GS_Windo
      w->UserPort))
nG    {
ay4    Class = Message->Class;
FS     Code = Message->Code;
6e     ReplyMsg (Message);
z1     switch (Class)
sL     {
iN5     case CLOSEWINDOW: {
OBP                       ausstieg ();
u3                        break;
OVN                      }
Hn5     case MOUSEBUTTONS: {
5qQ                        if ((!aufgebaut) && Code == SELECT
                           DOWN)
D9Z                               aufgebaut = baue_window_a
                                  uf ();
z8Q                        break;
5a0                       }
C35     case INACTIVEWINDOW: {
R9S                          if (aufgebaut) aufgebaut = baue_
                             window_ab ();
3C                           break;
9eQ                         }
fa5     case GADGETUP: {
orM                     aufgebaut = baue_window_ab ();
jd                      hole_screen ();
8H                      break;
EjK                    }
Fk4    }
G13   }
Hm1  }
InO }
(C) 1987 M&T
```

Listing. Kein Bildschirm bleibt mehr verborgen durch »GetScreen«. Bitte mit dem Checksummer (Seite 68) eingeben.

Ein Gang durch die Bibliotheken

Wissen Sie eigentlich, auf welche Bibliotheken Sie gerade zurückgreifen können? Ein kurzes C-Programm entlockt Ihrem Amiga dieses Geheimnis. Außerdem erfahren Sie noch ein paar Grundlagen über die Libraries und deren Aufbau.

Erst durch die Verwendung der Bibliotheksfunktionen des Amiga ist man in der Lage, diesen Computer wirklich auszunutzen. Doch dafür müssen erst die entsprechenden Libraries geöffnet werden, denn dadurch erhält man den Zugriff auf deren Befehle. Wollen Sie sehen, welche Libraries Ihnen zur Verfügung stehen? Dann sollten Sie das Listing abtippen und mit Ihrem C-Compiler übersetzen. Für Besitzer des Aztec-C-Compilers gelten folgende Aufrufe:

```
cc zeigelibs +l
ln zeigelibs.o -lc32
```

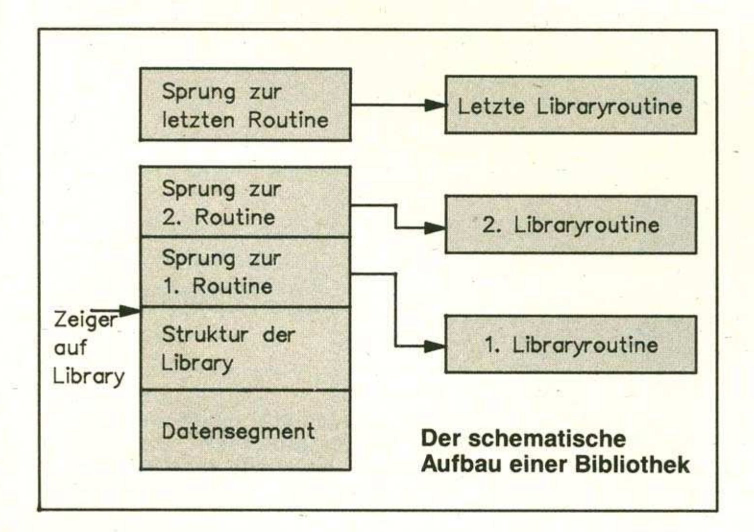
Für das Übersetzen mit dem Lattice-C-Compiler benutzen Sie die Befehle:

```
lc zeigelibs
blink lib:c.o,zeigelibs.o lib lib:lc.lib,lib:ami
ga.lib
```

Nach dem Starten des Programms mit dem Befehl »zeigelibs« wird für die voreingestellten Bibliotheken ausgegeben, ob Sie vorhanden sind. Wenn ja, erscheint auch noch die zugehörige Startadresse.

Zum Programm selbst gibt es nicht allzuviel zu sagen. In der for-Schleife (Zeilen 23 bis 37) werden zuerst die Namen der Bibliotheken zusammengesetzt. Dazu wird an die in den Zeilen 25 bis 27 vereinbarten Namen die Endung ».library« angehängt. Dann wird in Zeile 28 die entsprechende Library geöffnet. Der Ergebniswert »Base« ist ein »long«-Wert (32 Bit) und stellt einen Zeiger auf die Basis der Bibliothek dar. An dieser Stelle sei kurz einmal der allgemeine Aufbau einer Bibliothek erwähnt. Libraries bestehen grundsätzlich aus einer Struktur, in der wichtige Informationen zur Bibliothek stehen. Dahinter liegen eventuell nötige Daten. Vor dieser Struktur liegen Sprungbefehle zu den einzelnen Routinen. Die Routinen selbst haben keine bestimmte Position. Im Bild sehen Sie den schematischen Aufbau.

Es wird nur ein Zeiger definiert, da wir die Basisadresse nur kurz benötigen und die Library dann sofort wieder schließen. Wenn das Ergebnis von 0 verschieden ist, konnte die Library geöffnet werden und die Werte werden ausgegeben. Danach schließt das Programm sie sofort wieder (Zeile 33). Falls die Bibliothek nicht vorhanden ist, wird der Name und der Text »nicht da!« in Zeile 35 ausgegeben.



Der schematische Aufbau einer Bibliothek

Rufen Sie das Programm bitte nicht mehrmals hintereinander auf, da dies zu einer Guru-Meditation führen kann.

Lassen Sie ZeigeLibs einmal laufen, Sie werden erstaunt sein, auf wie viele Bibliotheken Sie bei der Programmierung zurückgreifen können. Und nur mit deren Routinen können Sie Ihren Amiga wirklich ausnutzen. (Kurt Hoffmann/rb)

Programmname:	ZeigeLibs
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4 oder Lattice-C V3.1
Aufrufe:	siehe Text

```
Programm : ZeigeLibs
--------------------

 1 os0 /* zeigelibs.c
 2 dP  #include <stdio.h> */
 3 80  #define VERSION 0
 4 B7  char *libname[15]={"exec","clist",
 5 Mg2   "graphics","layers","intuition",
 6 3C    "mathffp","mathieeedoubbas",
 7 LF    "mathtrans","dos","translator","icon",
 8 Ax    "diskfont","ramlib","info","version"};
 9 qC0 main()
10 6Z  {
11 ss1  char string[25];
12 oy   int loop;
13 1E   long Base;
14 1D   printf("\fzeigelibs by K.H.\n");
15 lo   printf(" zeigt verfuegbare Libraries:\n");
16 a2   printf("1.) ROM : je nach KICKSTART\n");
17 JC   printf("2.) LIBS: falls nicht im ROM,");
18 qe   printf(" sucht das System in LIBS:.\n");
19 Oy   printf("          Aenderung mit ASSIGN");
20 ii   printf(" ist moeglich.\n\n");
21 zI   printf("             Library-Name");
22 pA   printf("   Startadresse\n");
23 uh   for (loop=0;loop<15;loop++)
24 Kn   {
25 d13    *string='\0';
26 WC     strcpy(string,libname[loop]);
27 OX     strcat(string,".library");
28 Tz     Base=OpenLibrary(string,VERSION);
29 V3     if (Base != 0)
30 Qt     {
31 Zh4     printf("%24s :%7ld=%6lx",string,Base);
32 zE      printf(" (hex)\n");
33 CV      CloseLibrary(Base);
34 Y33    }
35 JJ     else printf("%24s : nicht da!\n",string);
36 Uq     Delay(50);
37 b61  }
38 c70 }
(C) 1988 M&T
```

Listing. Mit »ZeigeLibs« können Sie sehen, welche Bibliotheken zur Verfügung stehen.
Bitte mit dem Checksummer (Seite 68) eingeben.

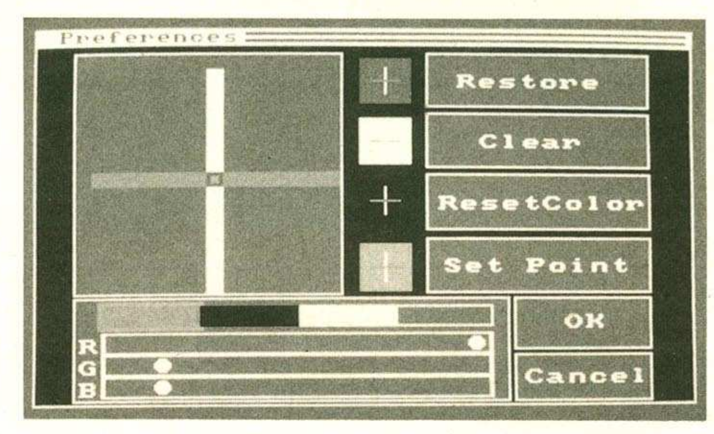
Mauszeiger ändern leicht gemacht

Um einen neuen Mauszeiger in C zu benutzen, benötigt man dessen Grafikdaten. Das Umrechnen per Hand ist mühselig und fehleranfällig. »PointerMaker« löst dieses Problem schnell und elegant mit Hilfe des Programms Preferences.

Bei dem Utility PointerMaker (siehe Listing 1) handelt es sich um ein Programm, das sicherlich für viele C-Programmierer von großem Nutzen sein kann. Der Teil des Amiga-Betriebssystems, der für die Darstellung von Fenstern und ähnliches zuständig ist, stellt die Möglichkeit bereit, jedem Fenster seinen individuellen Maus-Zeiger zuzuordnen. Der eine oder andere von Ihnen wird sicherlich schon von dieser Möglichkeit gehört haben, die zwar sehr praktisch, jedoch nur sehr schwer zu realisieren ist. Das Grafikbild für das Sprite mußte bisher immer von Hand in Hex-Zahlen umgerechnet werden.

Bevor PointerMaker benutzt wird, sollte man sich einen Entwurf des neuen Mauszeigers machen. Das Programm ist kein kompletter Editor für Mauszeiger, sondern eigentlich ein Konvertierungsprogramm. Es konvertiert den gerade aktuellen Mauszeiger in den entsprechenden Teil eines C-Quellprogramms. Zum Generieren des gewünschten Zeigers verwenden Sie einfach das Programm »Preferences«, das ja bei jedem Amiga mitgeliefert wird.

Dazu überträgt man mit Hilfe von Preferences (Menüpunkt »Edit Pointer«) den Zeiger. Hat man den Mauszeiger erstellt, verläßt man den Pointer-Editor mit »OK« und das Preferences Programm selbst mit »Use«. Sie sollten jetzt also Ihr »Kunstwerk«, den eben erstellten Mauszeiger, vor sich sehen. Nun starten Sie ein CLI-Window, sofern Sie sich noch nicht dort befinden sollten. Der Aufruf des Programms lautet wie folgt:



Mit Preferences erstellen Sie Ihren eigenen Mauszeiger

```
pointermaker datei
```

Dabei steht »datei« für den Namen der Datei und gleichzeitig für das Feld der Daten. Aus diesem Grund sollten Sie hier keinen Pfadnamen oder eine Namenserweiterung (zum Beispiel ».c«) verwenden, da sonst der C-Compiler beim Übersetzen Schwierigkeiten macht.

Nach Ablauf des Programmes haben Sie ein File mit dem von Ihnen eingegebenen Filenamen im aktuellen Directory. Dieses enthält die Definition der Sprite-Struktur als C-Source. Sie können den so erhaltenen Sourcecode leicht in eigene Programme einbauen, wie das kurze Demoprogramm »demo« (Listing 2) zeigt. Durch Klicken innerhalb und außerhalb des neuen Fensters ändert sich jedesmal der Mauszeiger. Dabei müssen Sie jedoch den Zeiger, den Sie entworfen haben, noch einmal mit Preferences ändern, um den Unterschied zu sehen. Wenn Sie die Daten in ein eigenes Programm einbauen wollen, vergessen Sie nie mit der Linkeroption »+Cdb« zu arbeiten, da sonst bei vorhandenem Fast-Memory kein Mauszeiger zu sehen ist.

Durch Eingabe von

```
pointermaker ?
```

erhalten Sie, wie von den CLI-Befehlen gewohnt, eine Angabe, wie der Aufruf auszusehen hat.

Das Programm selbst beinhaltet keine besonderen Tricks, aber es ist interessant, wie man aus der Preferences-Struktur Daten herausziehen kann. Dazu wird in Zeile 29 zuerst Speicher belegt, in den die Daten geschrieben werden. In der nächsten Zeile werden die Daten mit der Funktion »GetPrefs« geholt und abgelegt.

In den Zeilen 33 bis 37 werden dann der X- und Y-Offset des Hotpoints (Punkt der Maus, mit dem Sie Ihre Auswahl treffen), die Daten und der nötige Rest in die Datei geschrieben.

Vielleicht werden auch Sie demnächst Ihre eigenen Programme mit einem veränderlichen Maus-Zeiger ausstatten. Denn er kann den Benutzer auf Vorgänge hinweisen, die sonst mit der Ausgabe eines Textes angekündigt werden. Aber der neue Mauszeiger zerstört Ihre Bildschirmmaske nicht.

Außerdem ist es auch durchaus denkbar, sich bewegte Mauszeiger zu generieren ... (Ralf G. R. Bergs/rb)

Programmname:	PointerMaker
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4
Aufrufe:	cc pointermaker +C +D ln pointermaker +Cbd -lm -lc

```
Programm : PointerMaker
-----------------------

 1 ALO #include <stdio.h>
 2 cG  #include <functions.h>
 3 Rg  #include <intuition/intuition.h>
 4 1s  struct IntuitionBase   *IntuitionBase;
 5 Mp  struct Preferences   *prefs;
 6 Fb  FILE          *fpw;
 7 z0  done(how, why)
 8 Td  long   how;
 9 Po  char   *why;
10 6Z  {
11 iT3    if (fpw)
12 ba6       fclose(fpw);
13 Ug3    if (IntuitionBase)
14 V46       CloseLibrary(IntuitionBase);
15 FN3    fprintf(stderr, "%s\n", why);
16 6t     exit(how);
17 Hm0 }
18 gp  main(argc, argv)
19 yV  short                 argc;
20 Jn  char                 *argv[];
21 Hk  {
22 pD3    UBYTE                  i;
23 NO     if (argc != 2)
24 jM6       done(20L, "PointerMaker: Missing output filename");
25 TM3    if (*argv[1] == '?')
26 Dr6       done(0L, "\x9b4mUsage:\x9b0m \x9b7mPOINTERMAKER \"TO/A\
             "\n\x9b0;33;1mPointerMaker \x9b32m 1987 by Ralf G. R. B
             ergs\x9b0m");
27 7S3    if (! (IntuitionBase = (struct IntuitionBase *)   OpenLibr
          ary("intuition.library", 0L)))
28 t16       done(21L, "PointerMaker: Can't open intuition.library")
             ;
29 ay3    prefs = (struct Preferences *) AllocMem(232L, 0L);
30 BO     GetPrefs(prefs, 102L);
31 im     if (!(fpw = fopen(argv[1], "x")))
32 SP6       done(23L, "PointerMaker: Can't create output file");
33 aB3    fprintf(fpw, "/* XOffset = %d, YOffset = %d */\n\n", prefs
          ->XOffset, prefs->YOffset);
34 Uk     fprintf(fpw, "USHORT\t%s[] = {\n", argv[1]);
35 ec     for (i = 0; i < 35; i += 2)
36 aN6       fprintf(fpw, "\t0x%04x, 0x%04x,\n", prefs->PointerMatr
             ix[i], prefs->PointerMatrix[i+1]);
37 iM3    fprintf(fpw, "};\n");
38 Yj     if (ferror(fpw))
39 Z2     {
40 1A6       perror("PointerMaker");
41 Ki        done(24L, "");
42 gB3    }
43 9K     done(0L, "PointerMaker: Ok!");
44 iD0 }
(C) 1987 M&T
```

Listing 1. »PointerMaker« generiert Daten für neue, nach Ihren Wünschen gestaltete Mauszeiger. Bitte mit dem Checksummer (Seite 68) eingeben.

Programmname:	Demo
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4
Aufrufe:	cc demo +l ln demo.o -lc32 +Cdb

```
Programm : Demo
---------------

ALO #include <stdio.h>
Qf  #include <intuition/intuition.h>
Or  struct IntuitionBase   *IntuitionBase;
qT  struct GfxBase         *GfxBase;
K4  struct Window          *Window;
JG  struct NewWindow NewWindow = {
SS  50, 50, 200, 50, 0, 1, NULL, ACTIVATE, NULL, NULL,
6C  (UBYTE *)"Demo-Fenster", NULL, NULL, 0, 0,0 ,0, WBENCHSCREEN
SO  };
MI  /* Hier den von PointerMaker erzeugten Quellcode einfügen */
vJ  done(how,why)
1L  short   how;
Ts  char   *why;
Ad  {
nr3    if(Window)
Cf     {
5R6       ClearPointer(Window);
BC        CloseWindow(Window);
Jo3    }
Gu     if(IntuitionBase)
cB6       CloseLibrary(IntuitionBase);
Z53    fprintf(stderr,"%s\n",why);
wd     exit((long)how);
Ot0 }
ft  VOID main()
Mp  {
8a  LONG   i;
a73    if(!(IntuitionBase = (struct IntuitionBase *) OpenLibrary(
       "intuition.library",NULL)))
Cc6       done(20,"PointerMaker: Cant open intuition.library");
Xd3    if(!(Window = (struct Window *) OpenWindow(&NewWindow)))
fr6       done(21,"PointerMaker: Cant open window");
f63    SetPointer(Window,maus,16L,16L,-16L,-0L);
Vu0 /*                    ^              ^       ^
          Name des Zeigers   XOffset  YOffset */
TC3    Delay(3000);
Fe     done(0,"Demo: Ok!");
b60 }
(C) 1987 M&T
```

Listing 2. Ein Demoprogramm zu »Pointermaker«, das den neuen Mauszeiger auf den Bildschirm bringt. Bitte mit dem Checksummer (Seite 68) eingeben.

Entscheidungshilfe für Basic-Programmierer

Die sogenannten Requester können Ihnen zwar keine Entscheidung abnehmen, aber die Auswahl einer Alternative per Maus erleichtern. Mit unserem kleinen Basic-Unterprogramm sind Requester in Programmen in Zukunft ein Kinderspiel. Und zusätzlich wirkt Ihr Programm wesentlich besser als früher.

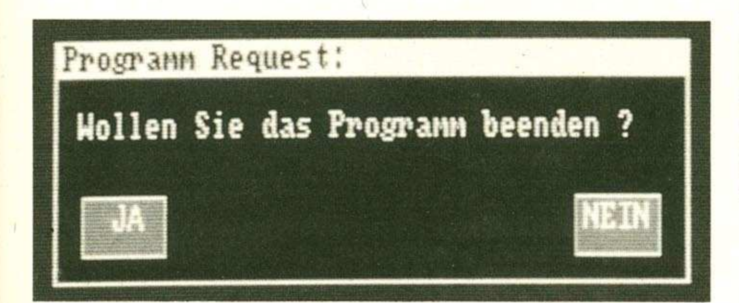
Wenn eine Entscheidung des Benutzers notwendig ist, werden oft Requester verwendet. Mit ihnen kann mit der Maus zwischen zwei Alternativen, normalerweise ja oder nein, gewählt werden. Doch wenn man für jede mögliche Abfrage einen eigenen Programmteil schreiben würde, wäre das gesamte Programm am Schluß sehr lang und unübersichtlich.

Die Lösung ist ein Unterprogramm, das flexibel auf verschiedene Übergabeparameter reagiert. Einer dieser Parameter ist der Text, mit dem der Benutzer informiert wird, welche Entscheidung er treffen soll. Der zweite und dritte Parameter sind die zwei Auswahlmöglichkeiten. Damit die Routine nicht zu lang wird, erscheint der Requester immer an derselben Stelle des Bildschirms. Ein Ändern des Programms, damit es auch verschiedene Positionen des Fensters erlaubt, ist jedoch völlig problemlos. Dies erreicht man durch die Änderung der Koordinaten bei den LINE- und LOCATE-Befehlen. Durch Experimentieren kommt man dabei recht schnell zu einem Ergebnis.

Wenn Sie das Programm verwenden, sind lediglich ein paar Punkte zu beachten. Der Aufruf sieht folgendermaßen aus:

```
CALL requester (x$,y$,z$)
```

Dabei ist x$ der Text, der den Benutzer informiert, im Beispiel (siehe Listing) »Programmabbruch ?«. Die Größe des Requesters stellt sich automatisch auf die Länge des Textes ein. Sollte dieser einmal weniger als 10 Zeichen lang sein, müssen Sie ihn durch Leerstellen davor und dahinter so längen, daß das Fenster nicht zu kurz wird.



So sieht ein Requester aus, der von dem Unterprogramm »Requester« auf den Bildschirm gebracht wird

y$ und z$ sind die Texte für die Schalter und dürfen nicht länger als vier Zeichen sein. Auch hier sind Änderungen natürlich kein Problem, verlängern aber die Routine erheblich.

Anstatt der Variablen können auch konstante Zeichenketten übergeben werden:

```
CALL requester ("Ein Test!","OK","Nein")
```

Soweit so gut, aber Sie benötigen ja auch noch ein Ergebnis. Dafür wird die Variable »req« verwendet, die als globale Variable im Unterprogramm definiert wird. Dadurch kann die Variable durch das Unterprogramm verändert werden und der neue Wert ist auch im Hauptprogramm bekannt. Wählen Sie die erste Alternative, erhalten Sie den Wert 1, sonst den Wert 2. Aufgrund der Variable req können Sie also nach Aufruf der Routine den weiteren Programmablauf bestimmen. Bei dem Demoprogramm erscheint so lange der Requester, bis der »JA«-Schalter angeklickt wird. Um das Unterprogramm in Ihren eigenen Programmen einsetzen zu können, löschen Sie einfach die Zeilen 23 bis 32. Vergessen Sie aber nicht, daß in Ihrem Hauptprogramm zur Auswertung die Variable mit dem Namen »req« verwendet wird.

Da das Programm kurz und übersichtlich ist, können Sie es nach eigenen Wünschen ändern und erhalten so eine Routine, die sich immer wieder sinnvoll einsetzen läßt.

Überhaupt sollten Sie bestrebt sein, sich durch solche kurze Routinen das Programmieren leichter zu machen. Die Fehlerrate wird durch das Verwenden ausgetesteter Programmteile nämlich deutlich geringer. Außerdem ersparen Sie sich natürlich ganz nebenbei noch viel Tipp- und Denkarbeit durch das Einsetzen solcher kurzen und nützlichen Routinen. (Erik Schmitt/rb)

```
Programm : Requester
--------------------

 1 ZfO '-------------Requester-Routine----------
 2 2W  SUB requester (a$,m1$,m2$) STATIC
 3 ER  SHARED req
 4 uo  WINDOW 3,"Programm Request:",(50,50)-(LEN(a$)*8+80,100),0
 5 bWA           CLS:COLOR 1,2:PAINT (4,4),2:PRINT:PRINT " "a$:COLOR
                  1,3
 6 in            LINE (10,30)-(50,30),1:LINE (10,45)-(50,45),1
 7 v0            LINE (10,30)-(10,45),1:LINE (50,45)-(50,30),1
 8 qI            PAINT (12,32),3,1
 9 hb            LOCATE 5,3:PRINT m1$
10 Gn            LINE (-20+8*LEN(a$),30)-(20+8*LEN(a$),30),1
11 7R            LINE (-20+8*LEN(a$),45)-(20+8*LEN(a$),45),1
12 ih            LINE (-20+8*LEN(a$),30)-(-20+8*LEN(a$),45),1
13 4n            LINE (20+8*LEN(a$),45)-(20+8*LEN(a$),30),1
14 6Q            PAINT (-10+8*LEN(a$),32),3,1
15 Mi            LOCATE 5,LEN(a$)-1:PRINT m2$
16 9m0 loop: WHILE MOUSE(0)<>1:WEND
17 yZ6       x=MOUSE(1):y=MOUSE(2)
18 fH        IF y<30 OR y>45 THEN loop
19 Kj        IF x>10 AND x<50 THEN req=1:WINDOW CLOSE 3
20 D2        IF x>-20+8*LEN(a$) AND x <20+8*LEN(a$) THEN req=2:WIN
             DOW CLOSE 3
21 NPO END SUB
22 M8  '------------Ende der Routine----------
23 tx  start:
24 9i3    x$="Programmabbruch ?"
25 ap     y$=" JA"
26 JO     z$="NEIN"
27 6IO schleife:
28 Bk3    CALL requester (x$,y$,z$)
29 54     LOCATE 18,2
30 Ob     IF req=2 THEN schleife
31 ZT     PRINT "Programmende"
32 GBO END
(C) 1988 M&T
```

Listing. Das Unterprogramm »Requester« mit kleinem Demoteil. Bitte mit dem Checksummer (Seite 68) eingeben.

Programmname:	Requester
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic 1.2

Ein Spiel wie aus dem Automaten

»Quatron« ist ein Spiel, das alle süchtig macht. Kein Wunder bei dieser Aufmachung und den vielfältigen Spielkombinationen. Lassen Sie sich überraschen, was Ihnen dieses kurze C-Programm an Witz und Spielfreude bietet.

Die Spielidee von Quatron kennen Sie vielleicht aus dem Film »Tron«. Mehrere Motorräder fahren auf der Spielfläche und hinterlassen eine Wand. Ziel ist es nun möglichst lange zu fahren, ohne gegen die wachsende Zahl von Mauern zu stoßen. Dabei ist es natürlich auch wichtig, die Gegner einzusperren, damit sie früher als man selbst einen Unfall haben. Der vordere Teile des Namens »Quatron« kommt von der maximalen Zahl von vier Motorrädern.

Das Programm kann man im CLI durch Eingabe des Namens oder auf der Workbench durch Anklicken starten. Es erscheint das Titelbild und man wird aufgefordert, die Anzahl der menschlichen Spieler einzugeben. Das sind alle, die vor dem Computer sitzen und begierig sind, Quatron zu spielen. Die Begierigen kann man wiederum daran erkennen, daß sich ihre Hande nervös am Joystick verkrampfen, oder daß sie sonst irgendwie unruhig reagieren. Entweder drückt man eine der Zifferntasten 0 bis 3 oder man klickt die auf dem Bildschirm stehende Zahl mit der Maus an.

Jetzt fragt der Computer nach der Anzahl der Computerspieler. Eine Null als Spieleranzahl interpretiert der Computer als Demospiel. Auch hier kann man wieder die Zahl direkt eingeben oder mit der Maus anklicken. Sie brauchen aber keine große Angst vor den Computerspielern haben. Diese sind zwar eine respektable Konkurrenz, aber nicht unschlagbar.

Dann geht's endlich los. Gespielt werden acht Runden. Die Steuerung mit den zwei Joysticks ist absolut. Egal in welche Richtung Ihr Motorrad gerade fährt, die Reaktion auf den Joystick bleibt immer gleich. Der dritte Spieler muß mit den Tasten (8, 4, 6, 2) des Ziffernblocks vorliebnehmen.

Ein Trick, den man seinem Mitspieler (= Gegner!) nicht verraten sollte, wenn man gewinnen will, ist, daß man per Knopfdruck (Joystick) oder <SPACE> (Tastatur) zwei bis vier Felder weit springen kann. So entkommt man vielleicht einer aussichtslos erscheinenden Situation.

Nach den acht Runden wird der Sieger angezeigt und die Frage gestellt, ob noch mal gespielt werden soll. Diese Frage wird (bis auf Notfälle) wohl meist durch Drücken von <j> oder anklicken des Wortes »Ja« beantwortet. Wenn Sie wirklich keine Zeit mehr haben, drücken Sie <n> oder klicken das Wort »Nein« an.

Bild. Dies ist die Fläche, auf der Sie gegen andere Spieler und den Computer antreten müssen

Es ist zu (fast) jeder Zeit möglich, das Spiel durch Drücken des unsichtbaren Close-Gadgets in der linken oberen Ecke des Spielfeldrahmens zu verlassen. Ebenso kann man eine einzelne Runde durch Drücken von <Esc> abbrechen und die nächste Runde beginnen.

Bei der Übersetzung des Programms muß durch den einzubindenden Assemblerteil einiges beachtet werden. Zuerst compilieren Sie den C-Teil (quatron.c) mit:

```
cc quatron
```

Danach übersetzen Sie die Assembler-Routine (myjoy.asm) durch den Aufruf:

```
as myjoy.asm
```

Nun können Sie beide Teile zusammenbinden:

```
ln quatron.o myjoy.o -lc
```

Wenn Sie allerdings eine Variante des Spiels vorziehen, bei der immer vier Motorräder im Spiel sind, löschen Sie einfach die Zeile 3. Danach compilieren Sie das Programm erneut mit den oben genannten Befehlen.

Durch die Verwendung des Assemblerteils ist es leider nicht möglich, das Programm mit dem Lattice-C-Compiler zu übersetzen, außer man ersetzt sie durch eine kleine C-Routine.

Quatron ist ein fantastisch gemachtes Spiel, das Sie sicher nicht so schnell loslassen wird.

Eine Möglichkeit ist zum Beispiel die Verwendung der Joystickroutine aus Ausgabe 12/87, Seite 106.

(Stefan Aust/Peter Miehle/rb)

Programmname:	Quatron
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4
Aufrufe:	siehe Text

```
Programm : Quatron
------------------

 1 u70 /*   Quatron
 2 yl  **   von Stefan M. Aust & P. Miehle */
 3 Sk  #define WUNSCH
 4 Sh  #include <intuition/intuition.h>
 5 fJ  #include <functions.h>
 6 Z2  struct IntuitionBase *IntuitionBase;
 7 tW  struct GfxBase *GfxBase;
 8 LC  struct Window *win;
 9 or  struct Screen *scr;
10 Si  struct RastPort *rp;
11 Nq  struct ViewPort *vp;
12 KB  short rnd(random)
13 G8  short random;
14 Ad  {
15 RQ2   static unsigned long wurzel = 26111967;
16 L6    if (random <= 0) wurzel = - random;
17 2I    else {
```

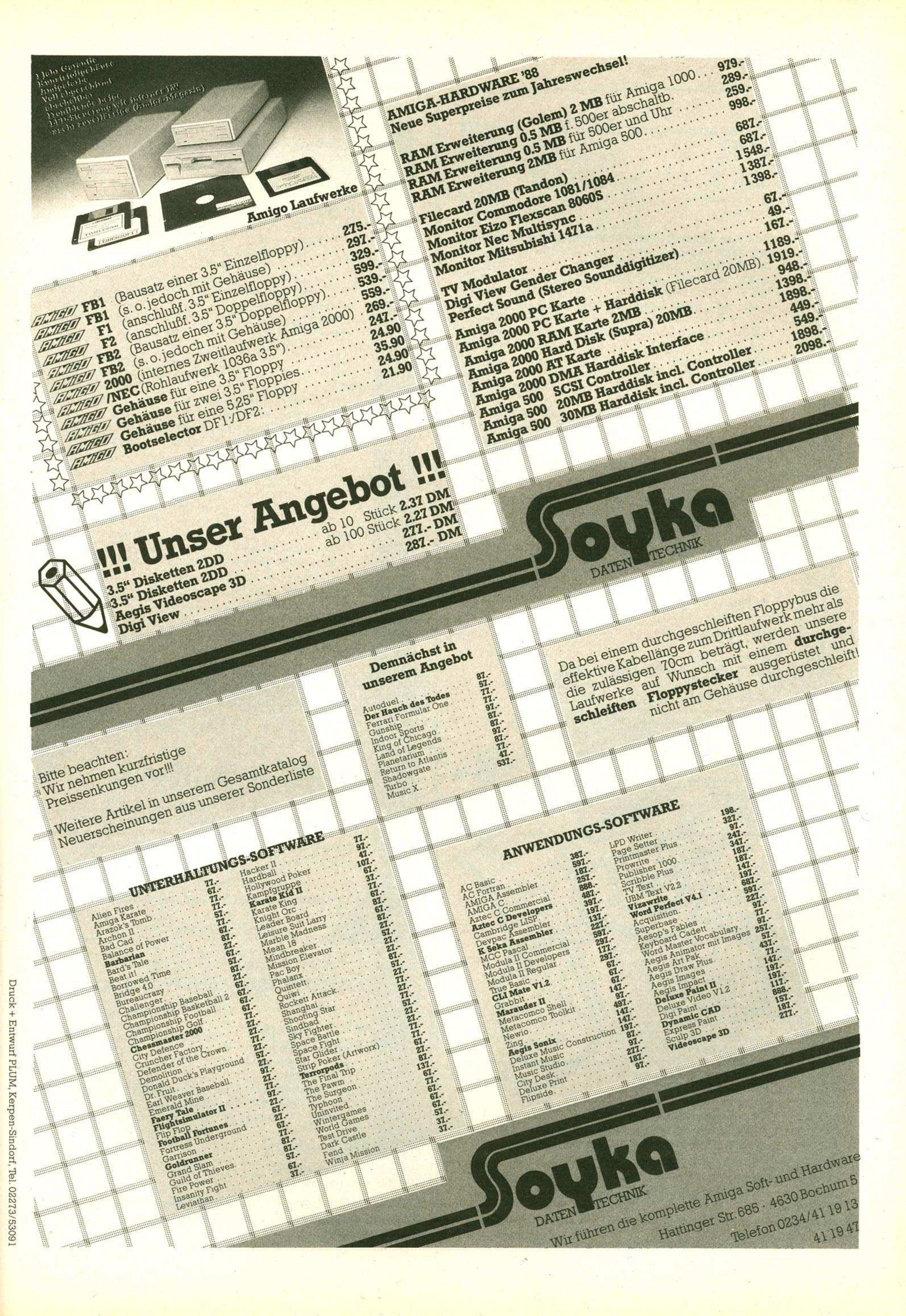
```
kf4      wurzel = (((wurzel * 4253261) & 0xFFFFFF) + 372837) & 0x
         FFFFFF;
c5       return ((unsigned short)wurzel % random);
Kp2    }
Lq0 }
zu  extern struct Joystick {
Xq2    LONG dx, dy; BOOL fire;
Iq0 } myjoy;
Fk  extern void myjoy0(), myjoy1();
Da  /* Funktionsübersicht: */
7J  void Vorbereitung(), Abbruch(/*short num*/);
xT  char *Zahl(/*short num,stellen*/);
yD  void Print(/*long x,y,farbe; char *str*/);
8C  void Box(/*long x1,y1,x2,y2,FgPen,BgPen; short breite*/);
X1  short Keytest(/*char *keystring*/);
D9  void Hintergrund(), Vorher(), Nachher();
tb  void Titel(/*short vorher*/);
1S  void TestMaus(/*short vorher*/), TestKey(/*short vorher*/);
XS  void Runde(/*short runde*/);
pm  void SetBlock(/*short spieler; long x,y*/);
Cz  short TestBlock(/*long x,y*/);
sh  void BonusPunkte(/*short anzahl*/);
Jy  void Drive();
hX  void Sprung(/*struct Spieler *hops*/);
wW  /* globale Variablen: */
TX  struct Spieler {
w92    long x, y;
hM     long r;
Ue     short count;
tL     BOOL crash, mensch;
YC     BOOL sprung;
dh0 } sp[4];
Ry  long xr[] = { 0, -4, 0, 4 };
gn  long yr[] = { -4, 0, 4, 0 };
mU  USHORT key;
Ti  BOOL ende, quit, weiter;
tB  short menschlich = 0;
dZ  #ifdef WUNSCH
zI  short compulich = 0;
Jo  short maxcompu;
C8  #endif
sV  struct TextAttr txtAttr = {
NW2    (STRPTR)"topaz.font",
zQ     TOPAZ_EIGHTY,
dn     FS_NORMAL,
zV     FPF_ROMFONT
KG0 };
RL  struct NewScreen newscr = {
mt2    0, 0, 320, 222,
Pt     5, 1, 0,
et     0, CUSTOMSCREEN,
U1     &txtAttr, (UBYTE *)"QUATRON / V1.5 / 14.11.87",
As     NULL, NULL
RN0 };
3d  struct NewWindow newwin = {
o62    0, 10, 320, 210, 1, 0,
6X     CLOSEWINDOW| RAWKEY| MOUSEBUTTONS,
ox     WINDOWCLOSE| SMART_REFRESH| ACTIVATE,
9i     NULL, NULL, NULL,
b3     NULL, NULL, 0,0,0,0,
B6     CUSTOMSCREEN
ZV0 };
EF  /* Alle Initialisierungen stehen in dieser Routine.
gw  ** Sollte ein Fehler auftreten, so wird das Programm sofort
T7  ** ordungsgemäß beendet. (siehe nächstes UP)*/
r2  void Vorbereitung()
Hk  {
Oz2    static short neueFarben[] = {
z0     0xC0C, 0xFBF,
uf     0xE00, 0xF80,
vn     0x9AA, 0x677,
AJ     0x4F0, 0x4A0,
yk     0x08F, 0x05A,
N1     0xFE0, 0xB90,
Zn     0x0EF, 0x0BC,
U0     0x000, 0x000
ok     };
F8     if ((IntuitionBase = (struct IntuitionBase *)
bo3     OpenLibrary("intuition.library",0L)) == NULL) Abbruch(1);

D02    if ((GfxBase = (struct GfxBase *)
Qy3     OpenLibrary("graphics.library",0L)) == NULL) Abbruch(2);
X62    if ((scr = (struct Screen *) OpenScreen(&newscr)) == NULL)
        Abbruch(3);
w7     newwin.Screen = scr;
Hy     if ((win = (struct Window *) OpenWindow(&newwin)) == NULL)
        Abbruch(4);
XJ     LoadRGB4(&scr->ViewPort, neueFarben, (long)sizeof(neueFar
       ben)/2);
e90 }
5e  /* Diese Routine bricht das Programm ab und de-initialisiert
     schon
gs  ** initialisierte Funktionen.
IF  ** Wird der Routine eine Zahl ungleich Null übergeben, so ge
    ht
Q7  ** sie von einem Fehler aus und zeigt diese Fehlernummer an.
     */
TR  void Abbruch(num)
aq  register short num;
hA  {
wa2    static BYTE alertStr[] = "\0\24\14ABBRUCH - FEHLER #000000
       00\0\1\1\254\14[BITTE MAUSTASE DRÜCKEN]\0\0";
jI     if (win) CloseWindow(win);
6S     if (scr) CloseScreen(scr);
uI     if (GfxBase) CloseLibrary(GfxBase);
vF     if (IntuitionBase) {
SJ4       if (num) {  alertStr[28] = (num%10)+'0';
cG6          alertStr[27] = (num/10)+'0';
Mq           DisplayAlert(RECOVERY_ALERT, alertStr, 20L);
uP4       }
C1        CloseLibrary(IntuitionBase);
wR2    }
Om     exit(0);
yT0 }
ib  char *Zahl(num, stellen)
q6  register short num;
V0  register short stellen;
yR  {
nr2    static char ziff[10];
qr     ziff[stellen] = '\0';
RH     do {
un4       ziff[--stellen] = num % 10 + '0';
6G        num /= 10;
8d2    }
cE     while (stellen>0);
7x     return (ziff);
Bg0 }
97  void Print(x, y, farbe, str)
75  long x, y, farbe;
fi  char *str;
Be  {
HM2    BYTE APen = rp->FgPen;
A0     SetAPen(rp, farbe);
77     Move(rp, x, y+rp->TxBaseline); Text(rp, str, (long)strlen
       (str));
Aw     SetAPen(rp, (long)APen);
Kp0 }
NE  void Box(x1, y1, x2, y2, FgPen, BgPen, breite)
pK  register long x1, y1, x2, y2; long FgPen, BgPen;
He  short breite;
Kn  {
8h2    register short i;
RW     BYTE APen = rp->FgPen;
jr     SetAPen(rp, BgPen);
9f     RectFill(rp, x1, y1, x2, y2);
1D     SetAPen(rp, FgPen);
AV     for (i=0; i<breite; i++) {
G74       Move(rp, x1+i, y1+i);
aT        Draw(rp, x2-i, y1+i); Draw(rp, x2-i, y2-i);
96        Draw(rp, x1+i, y2-i); Draw(rp, x1+i, y1+i);
Y32    }
QC     SetAPen(rp, (long)APen);
a50 }
2J  short Keytest(keystring)
W7  register char *keystring;
Z2  {
J62    register short i, k;
JM     i = 0;
gg     while (k = *keystring++) {
TR4       i++;
3N        if (k == key) {key = 0; return(i); }
```

Listing 1. »Quatron«, ein gut gemachtes, packendes Spiel. Bitte mit dem Checksummer (Seite 68) eingeben.

```
jE2    }
D8     return(0);
1G0  }
mw   /* Hier beginnt das "Master Control Program". */
Uq   main()
kD   {
vb2    register short i, d;
9t     static char sp_text[] = "SP0:";
ih     long now[3];
Ej     Vorbereitung();
z6     DateStamp(&now);
de     rnd(-(int)(now[0]^now[1]^now[2]));
jc     rp = win->RPort; vp = &scr->ViewPort;
QQ     quit = weiter = FALSE;
T9     Box(0L, 0L, 319L, 199L, 3L, 0L, 2);
UE     Box(2L, 2L, 317L, 197L, 2L, 0L, 2);
LB     do {
Re4      ende = FALSE;
RM       Hintergrund();
Qz       Titel(1);
FU       if (quit) Abbruch(0);
pl0  #ifdef WUNSCH
e54      maxcompu = 1;
Xi       switch(menschlich) {
wt       case 0: compulich=4;
3C7         break;
yJ4      case 1: maxcompu++;
1N       case 2: maxcompu++;
Ha       case 3:
dq6        ende = FALSE;
dY         Hintergrund();
iJ         Titel(3);
Rg         if (quit) Abbruch(0);
BK         break;
Hm4      }
ZV0  #endif
1T4      SetDrMd(rp, JAM2); SetBPen(rp, 0L);
Hr       for (i=0; i<4; i++) sp[i].count = 0;
vm       Print(0L, 203L, 1L," SP1:0000  SP2:0000  SP3:0000  SP4:0000 ");
ga       for (d=0; d<8; d++) {
bU6        Runde(d);
VG         sp[0].x = 312; sp[0].y = 192;
Xg         sp[1].x = 312; sp[1].y = 4;
e0         sp[2].x = 4; sp[2].y = 4;
Fp         sp[3].x = 4; sp[3].y = 192;
05         for (i=0; i<4; i++) {
2U8          sp[i].r = i;
FB0  #ifdef WUNSCH
f78          sp[i].crash = (i >= (menschlich+compulich));
EV0  #else
t88          sp[i].crash = FALSE;
pl0  #endif
P08          sp[i].mensch = (i < menschlich);
kF           sp_text[2] = i+'1';
Cg3       Print(8L+i*80L, 203L, 1L, sp_text);
c76        }
1M         BonusPunkte(menschlich*8);
5I         ende = FALSE;
YH         Drive();
s7         if (quit) Abbruch(0);
Y4         Delay(50L);
iD4      }
F8       Delay(100L);
B0       ende = FALSE;
B6       Hintergrund();
Dn       Titel(2);
zE       if (quit) Abbruch(0);
oJ2    }
WS     while(weiter);
ON     Abbruch(0);
rM0  }
uK   void Hintergrund()
pI   {
1q2    register long i,j;
8K     register BOOL check = TRUE;
Ut     for (i=4; i<=308; i+=8) {
GA4      for (j=4; j<=188; j+=8) {
v36        if (check) SetAPen(rp, 4L); else SetAPen(rp, 5L);
Mw         check = !check;
qX         RectFill(rp, i, j, i+7L, j+7L);
1W4      }
Pz       check = !check;
3Y2    }
4Z0  }
pX   void Vorher()
2V   {
8p2    register long i;
ps     SetDrMd(rp, JAM1);
yy     Box(50L, 120L, 269L, 142L, 1L, 0L, 2);
Xy     Print(60L, 124L, 1L, "WIEVIELE MENSCHEN MÖCHTEN");
q3     Print(68L, 132L, 1L, "AN QUATRON TEILNEHMEN ?");
qJ     Box(184L, 152L, 207L, 175L, 1L, 0L, 2);
R0     Box(74L, 160L, 117L, 174L, 1L, 0L, 2);
3i     Print(80L, 164L, 1L, "0123");
b1     SetAPen(rp, 1L);
gj     for (i=0; i<2; i++) {
5T4      Move(rp, 91L+i, 143L);
bx       Draw(rp, 91L+i, 159L);
Mk       Move(rp, 163L+i, 143L);
5V       Draw(rp, 163L+i, 164L-i);
Bv       Draw(rp, 184L, 164L-i);
Mr2    }
Ns0  }
95   #ifdef WUNSCH
Hj   void ZwDurch()
Mp   {
S92    register long i;
Kk     static char anzahl[] = "0123";
II     Box(50L, 120L, 269L, 142L, 1L, 0L, 2);
70     Print(64L, 124L, 1L, "UND WIEVIELE SPIELER DES");
le     Print(56L, 132L, 1L, "COMPUTERS SIND ZUGELASSEN?");
A3     Box(184L, 152L, 207L, 175L, 1L, 0L, 2);
li     Box(74L, 160L, 117L, 174L, 1L, 0L, 2);
4J     anzahl[maxcompu+1] = '\0';
A2     Print(80L, 164L, 1L, anzahl);
wM     SetAPen(rp, 1L);
14     for (i=0; i<2; i++) {
Qo4      Move(rp, 91L+i, 143L);
wI       Draw(rp, 91L+i, 159L);
h5       Move(rp, 163L+i, 143L);
Qq       Draw(rp, 163L+i, 164L-i);
WG       Draw(rp, 184L, 164L-i);
hC2    }
iD0  }
Ow   #endif
hj   void Nachher()
hA   {
V42    register short i;
rK     register short sieger = 0;
rz     static char sieger1[] = "SIEGER IST SPIELER X",
Yb9           sieger2[] = "MIT 0000 PUNKTEN !";
yW2    char *src, *dest;
XA     for (i=1; i<4; i++)
zM4      if (sp[i].count > sp[sieger].count) sieger = i;
wZ2    sieger1[19] = sieger + '1';
Ds     src = Zahl(sp[sieger].count, 4); dest = &sieger2[4];
Pw     for (i=0; i<4; i++) *dest++ = *src++;
1t     Box(70L, 112L, 249L, 134L, 1L, 0L, 2);
h9     Print(80L, 116L, 1L, sieger1);
Cm     Print(88L, 124L, 1L, sieger2);
TS     Box(70L, 145L, 153L, 159L, 1L, 0L, 2);
1u     Print(80L, 149L, 1L, "NOCHMAL?");
D3     Box(208L, 160L, 231L, 183L, 1L, 0L, 2);
KU     Box(130L, 169L, 198L, 183L, 1L, 0L, 2);
YG     Print(136L, 173L, 1L, "JA NEIN");
Qq     SetAPen(rp, 1L);
VY     for (i=0; i<2; i++) {
Ek4      Move(rp, 98L+i, 135L);
BY       Draw(rp, 98L+i, 144L);
po       Move(rp, 154L, 147L+i);
vw       Draw(rp, 220L-i, 147L+i);
oX       Draw(rp, 220L-i, 159L);
nl       Move(rp, 154L, 155L+i);
1M       Draw(rp, 172L-i, 155L+i);
0z       Draw(rp, 172L-i, 168L);
Ej2    }
Fk0  }
mR   #define Q_START_X    4
K0   #define Q_START_Y    84
dx   #define REST_START_X (Q_START_X+16*4)
```

Listing 1. »Quatron«, ein gut gemachtes, packendes Spiel. Bitte mit dem Checksummer (Seite 68) eingeben. (Fortsetzung)

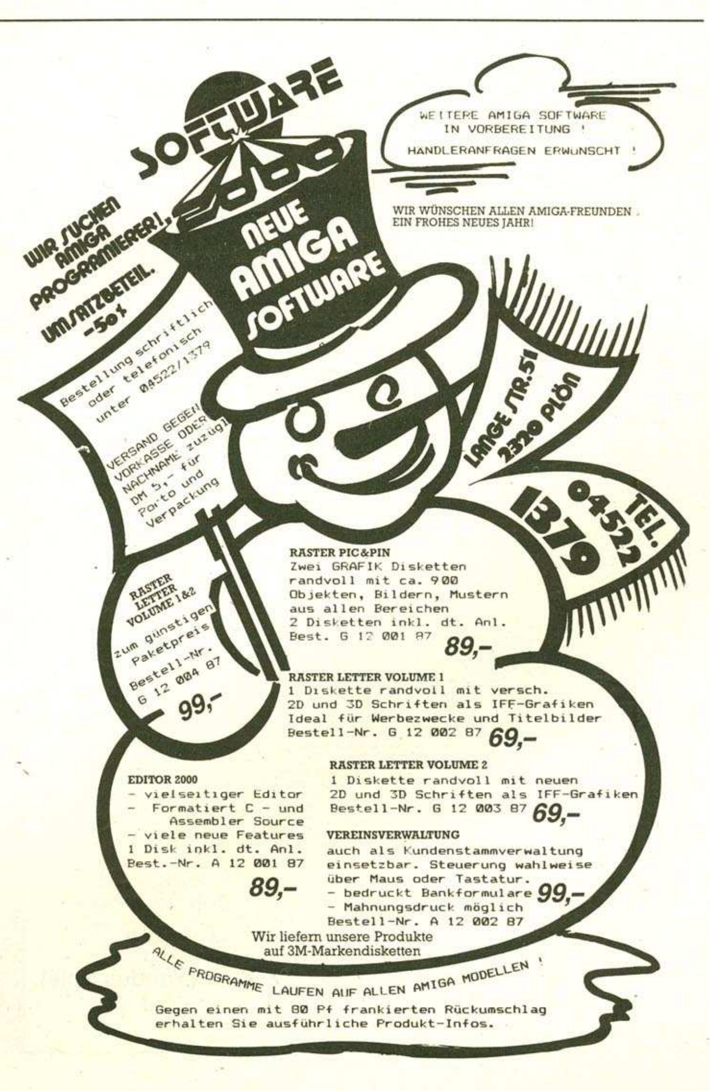

Amigo Laufwerke
AMIGO FB1 (Bausatz einer 3.5" Einzelfloppy) 275.-
AMIGO FB1 (s. o. jedoch mit Gehäuse) 297.-
AMIGO F1 (anschlußf. 3.5" Einzelfloppy) 329.-
AMIGO F2 (anschlußf. 3.5" Doppelfloppy) 599.-
AMIGO FB2 (Bausatz einer 3.5" Doppelfloppy) 539.-
AMIGO FB2 (s. o. jedoch mit Gehäuse) 559.-
AMIGO 2000 (internes Zweitlaufwerk Amiga 2000) 269.-
AMIGO /NEC (Rohlaufwerk 1036a 3.5") 247.-
AMIGO Gehäuse für eine 3.5" Floppy 24.90
AMIGO Gehäuse für zwei 3.5" Floppies 35.90
AMIGO Gehäuse für eine 5,25" Floppy 24.90
AMIGO Bootselector DF1:/DF2: 21.90
AMIGA-HARDWARE '88
Neue Superpreise zum Jahreswechsel!
RAM Erweiterung (Golem) 2 MB für Amiga 1000
RAM Erweiterung 0.5 MB f. 500er abschaltb.
RAM Erweiterung 0.5 MB für 500er und Uhr
RAM Erweiterung 2MB für Amiga 500
Filecard 20MB (Tandon)
Monitor Commodore 1081/1084
Monitor Eizo Flexscan 8060S
Monitor Nec Multisync
Monitor Mitsubishi 1471a
TV Modulator
Digi View Gender Changer
Perfect Sound (Stereo Sounddigitizer)
Amiga 2000 PC Karte
Amiga 2000 PC Karte + Harddisk (Filecard 20MB)
Amiga 2000 RAM Karte 2MB
Amiga 2000 Hard Disk (Supra) 20MB
Amiga 2000 AT Karte
Amiga 2000 DMA Harddisk Interface
Amiga 500 SCSI Controller
Amiga 500 20MB Harddisk incl. Controller
Amiga 500 30MB Harddisk incl. Controller
979.- 289.- 259.- 998.- 687.- 687.- 1548.- 1387.- 1398.- 67.- 49.- 167.- 1189.- 1919.- 948.- 1398.- 1898.- 449.- 549.- 1898.- 2098.-
!!! Unser Angebot !!!
3.5" Disketten 2DD ab 10 Stück 2.37 DM
3.5" Disketten 2DD ab 100 Stück 2.27 DM
Aegis Videoscape 3D 277.- DM
Digi View 287.- DM
Soyka
DATEN TECHNIK
Da bei einem durchgeschleiften Floppybus die effektive Kabellänge zum Drittlaufwerk mehr als die zulässigen 70cm beträgt, werden unsere Laufwerke auf Wunsch mit einem durchgeschleiften Floppystecker ausgerüstet und nicht am Gehäuse durchgeschleift!
Demnächst in unserem Angebot
Autoduel
Der Hauch des Todes
Ferrari Formular One
Gunship
Indoor Sports
King of Chicago
Land of Legends
Planetarium
Return to Atlantis
Shadowgate
Turbo
Music X
87.- 57.- 77.- 77.- 97.- 87.- 87.- 97.- 87.- 77.- 47.- 537.-
Bitte beachten:
Wir nehmen kurzfristige Preissenkungen vor!!!
Weitere Artikel in unserem Gesamtkatalog
Neuerscheinungen aus unserer Sonderliste
UNTERHALTUNGS-SOFTWARE
Alien Fires
Amiga Karate
Arazok's Tomb
Archon II
Bad Cad
Balance of Power
Barbarian
Bard's Tale
Beat it!
Borrowed Time
Bridge 4.0
Bureaucrazy
Challenger
Championship Baseball
Championship Basketball 2
Championship Football
Championship Golf
Chessmaster 2000
City Defence
Cruncher Factory
Defender of the Crown
Demolition
Donald Duck's Playground
Dr. Fruit
Earl Weaver Baseball
Emerald Mine
Faery Tale
Flightsimulator II
Flip Flop
Football Fortunes
Fortress Underground
Garrison
Goldrunner
Grand Slam
Guild of Thieves
Fire Power
Insanity Fight
Leviathan
77.- 67.- 77.- 77.- 57.- 77.- 67.- 87.- 27.- 67.- 57.- 87.- 27.- 67.- 67.- 67.- 67.- 77.- 27.- 27.- 77.- 27.- 57.- 27.- 97.- 27.- 77.- 97.- 27.- 67.- 27.- 67.- 77.- 87.- 87.- 57.- 67.- 37.-
Hacker II
Hardball
Hollywood Poker
Kampfgruppe
Karate Kid II
Karate King
Knight Orc
Leader Board
Leisure Suit Larry
Marble Madness
Mean 18
Mindbreaker
Mission Elevator
Pac Boy
Phalanx
Quintett
Quiwi
Rockett Attack
Shanghai
Shooting Star
Sindbad
Sky Fighter
Space Battle
Space Fight
Star Glider
Strip Poker (Artworx)
Terrorpods
The Final Trip
The Pawm
The Surgeon
Typhoon
Uninvited
Wintergames
World Games
Test Drive
Dark Castle
Fend
Winja Mission
77.- 97.- 47.- 107.- 67.- 37.- 77.- 67.- 87.- 67.- 87.- 27.- 57.- 27.- 27.- 87.- 57.- 27.- 77.- 27.- 77.- 57.- 27.- 27.- 77.- 77.- 67.- 37.- 81.- 137.- 67.- 77.- 67.- 67.- 67.- 57.- 37.- 37.-
ANWENDUNGS-SOFTWARE
AC Basic
AC Fortran
AMIGA Assembler
AMIGA C
Aztec C Commercial
Aztec C Developers
Cambridge LISP
Devpac Assembler
K Seka Assembler
MCC Pascal
Modula II Commercial
Modula II Developers
Modula II Regular
True Basic
CLI Mate V1.2
Grabbit
Marauder II
Metacomco Shell
Metacomco Toolkit
Newio
Zing
Aegis Sonix
Deluxe Music Construction
Instant Music
Music Studio
City Desk
Deluxe Print
Flipside
387.- 597.- 187.- 257.- 888.- 487.- 397.- 197.- 137.- 227.- 597.- 297.- 177.- 297.- 67.- 67.- 67.- 147.- 97.- 497.- 147.- 147.- 197.- 87.- 97.- 277.- 187.- 97.-
LPD Writer
Page Setter
Printmaster Plus
Prowrite
Publisher 1000
Scribble Plus
TV Text
UBM Text V2.2
Vizawrite
Word Perfect V4.1
Acquisition
Superbase
Aesop's Fables
Keyboard Cadett
Word Master Vocabulary
Aegis Animator mit Images
Aegis Art Pak
Aegis Draw Plus
Aegis Images
Aegis Impact
Deluxe Paint II
Deluxe Video V1.2
Digi Paint
Dynamic CAD
Express Paint
Sculp 3D
Videoscape 3D
198.- 327.- 97.- 247.- 347.- 187.- 187.- 147.- 197.- 687.- 597.- 227.- 97.- 77.- 97.- 257.- 57.- 437.- 77.- 147.- 197.- 197.- 117.- 888.- 157.- 187.- 277.-
Soyka
DATEN TECHNIK
Wir führen die komplette Amiga Soft- und Hardware
Hattinger Str. 685 · 4630 Bochum 5
Telefon 0234/41 19 13
41 19 47
Druck + Entwurf PLUM, Kerpen-Sindorf, Tel. 02273/53091

```
T1   #define REST_START_Y (Q_START_Y-7*4)
5A   #define R0 0
mD   #define R1 64
S6   #define R2 128
X8   #define R3 192
bA   void Titel(vorher)
B4   short vorher;
Mp   {
Z02    static char autoren[] = "VON ST.M.AUST UND P.MIEHLE";
Du     static UBYTE Daten[] = {
yF4      255, Q_START_X, Q_START_Y,
ZU       13+R0, 1+R3,13+R2, 1+R3,13+R0, 1+R3,13+R2, 1+R3, 1+R0, 1+R3,
gd5      1+R2, 1+R3, 1+R0, 1+R3, 1+R2, 3+R3, 2+R2, 5+R3, 1+R0, 2+R1,
hM       8+R0, 1+R1, 8+R2, 1+R1,14+R0, 7+R1, 1+R2, 6+R3,11+R2, 2+R1,
tH4      10+R0, 1+R3, 9+R2,
P9       255, REST_START_X, REST_START_Y,
TB5      7+R2,35+R3, 6+R0, 5+R3, 6+R2, 9+R3, 6+R0, 3+R3, 6+R2, 1+R3,
zC       6+R0, 5+R3, 6+R2, 3+R3, 5+R0, 1+R1, 4+R2, 1+R1, 6+R0,11+R1,
pY       6+R2, 7+R1, 5+R0, 5+R3, 4+R2, 1+R3, 5+R0,13+R1, 6+R2, 1+R1,
Yg       6+R0, 1+R1, 6+R2, 3+R1, 6+R0, 1+R1, 6+R2, 1+R1, 9+R0,34+R3,
sJ       1+R0,34+R1, 2+R0, 1+R1, 2+R2,26+R1, 1+R2,26+R3, 9+R2, 5+R1,
br       6+R0, 9+R1, 5+R2, 1+R3, 4+R0, 7+R3, 5+R2, 1+R1, 4+R0, 1+R1,
51       4+R2, 9+R1, 6+R0, 1+R1, 6+R2, 1+R1, 6+R0, 1+R1, 6+R2, 5+R1,
NS       6+R0, 0
Ow2    };
wS     UBYTE data;
xX     register UBYTE *daten;
e0     register long x, y;
sL     short farbe;
wy     short i, r, len;
6q     struct IntuiMessage *Nachricht;
RS     ULONG Art;
WZ     SetDrMd(rp, JAM1);
bX0  #ifdef WUNSCH
112    if (vorher == 3) ZwDurch(); else
950  #endif
Ne2    if (vorher == 1) Vorher(); else Nachher();
NC     farbe = rnd(4);
eF     daten = Daten;
Nk     while(!ende) {
IR4      if ((data = *daten++) == 0) {
RG6        farbe = rnd(4);
iJ         daten = Daten;
7Y         continue;
1W4      }
98       if (data == 255) {
da6        x = (long)*daten++;
ge         y = (long)*daten++;
vG         SetBlock(farbe, x, y);
t0         Print(76L, 92L, farbe*2+7L, autoren);
w9         Print(77L, 93L, farbe*2+6L, autoren);
8d4      }
uA       else {
JT6        r = (short)data>>6;
pn         len = (short)data%64;
Ju         for (i=0; i<len; i++) {
6g8          x += xr[r];
Bn           y += yr[r];
5Q           SetBlock(farbe, x, y);
OU3     if (Nachricht = (struct IntuiMessage *) GetMsg(win->UserPort)) {
k25       Art = Nachricht->Class;
RZ        key = Nachricht->Code;
NL        ReplyMsg(Nachricht);
JX        if (Art == CLOSEWINDOW) ende = quit = TRUE;
JY        if (Art == MOUSEBUTTONS) TestMaus(vorher);
i2        if (Art == RAWKEY) TestKey(vorher);
Ns3     }
xi      Delay(2L);
Pu6       }
Qv4     }
Rw2  }
Sx0  }
Fz   void TestMaus(vorher)
sH   register short vorher;
Ru   {
jR2    register short mx, my;
m4     char m;
CE     if (key == SELECTDOWN) {
KG0  #ifdef WUNSCH
E14      if (vorher == 3) {
X56        mx = (win->MouseX-80)/8;
KY         my = (win->MouseY-168)/4;
dA         if (mx>=0 && mx<=maxcompu && my==0) {
0R8          compulich = mx;
073      m = compulich + '0';
tm       Print(192L, 160L, 1L, &m);
bu       ende = TRUE;
iD6        }
jE4      }
ZM       else
2y0  #endif
E94      if (vorher == 1) {
jH6        mx = (win->MouseX-80)/8;
Wk         my = (win->MouseY-168)/4;
gH         if (mx>=0 && mx<=3 && my==0) {
x38          menschlich = mx;
9D           m = menschlich + '0';
5y           Print(192L, 160L, 1L, &m);
n6           ende = TRUE;
uP6        }
vQ4      }
hx       else {
Xx6        mx = (win->MouseX-136)/8;
kt         my = (win->MouseY-177)/4;
2g         if (mx>=0 && mx<=6 && my==0) {
EG8          if (mx>2) m = 'N'; else m = 'J';
PY3      weiter = (m=='J');
sC       Print(216L, 168L, 1L, &m);
xG       ende = TRUE;
4Z6        }
5a4      }
6b2    }
7c0  }
v2   void TestKey(vorher)
Xw   register short vorher;
6Z   {
wW2    register short j;
Rj     char m;
yu0  #ifdef WUNSCH
sf2    if (vorher == 3) {
jt4      if (j = Keytest("\12\1\2\3")) {
rS6        compulich = --j;
CI         if (compulich <= maxcompu) {
dk8          m = compulich + '0';
WP3     Print(192L, 160L, 1L, &m);
EX      ende = TRUE;
Lq6        }
Mr4      }
Ns2    }
D0     else
gc0  #endif
sn2    if (vorher == 1) {
v54      if (j = Keytest("\12\1\2\3")) {
y46        menschlich = --j;
lp         m = menschlich + '0';
ha         Print(192L, 160L, 1L, &m);
Pi         ende = TRUE;
W14      }
X22    }
JZ     else {
SQ4      if (j = Keytest("\46\66")) {
b16        m = 70+j*4;
z8         weiter = (m=='J');
Sm         Print(216L, 168L, 1L, &m);
Xq         ende = TRUE;
e94      }
fA2    }
gB0  }
ec   void Runde(runde)
ba   register short runde;
```

Listing 1. »Quatron«, ein gut gemachtes, packendes Spiel. Bitte mit dem Checksummer (Seite 68) eingeben. (Fortsetzung)

```
f8  {
oF2     static char r_text[] = "RUNDE 0";
B6      Hintergrund();
N5      Box(112L, 86L, 207L, 112L, 3L, 0L, 2);
Ev      Box(114L, 88L, 205L, 110L, 2L, 0L, 2);
WN      r_text[6] = runde+'1';
rF      Print(132L, 96L, 1L, r_text);
hD      Delay(50L);
HC      Hintergrund();
sNO }
EX  void SetBlock(spieler, x, y)
u2  short spieler;
pB  register long x, y;
sL  {
CW2     register long f1, f2;
06      if (spieler > 5) { f1 = f2 = spieler*2L-5L; }
noE                 else { f1 = f2 = spieler*2L+6L; f2++; }
gX2     SetAPen(rp, f2);
Hb      RectFill(rp, x, y, x+3L, y+3L);
dT      SetAPen(rp, f1);
CU      RectFill(rp, x, y, x+2L, y+2L);
4Z0 }
ba  short TestBlock(x, y)
LY  long x, y;
3W  {
1G2     register short farbe;
hV      farbe = ReadPixel(rp, x, y);
FQ      if (farbe<4) return(99);
NU      if (farbe>13) return(-1);
oU      if (farbe>5) return(farbe/2-2);
gb      return(0);
EjO }
1J  void BonusPunkte(anzahl)
jn  short anzahl;
Dg  {
BO2     register long tx, ty;
eD      while (anzahl--) {
ia4        tx = rnd(75)*4 + 12L;
UZ         ty = rnd(45)*4 + 12L;
HO         SetBlock(4, tx, ty);
GZ         SetBlock(4, tx+4, ty);
1Z         SetBlock(4, tx+4, ty+4);
PC         SetBlock(4, tx, ty+4);
Qv2     }
RwO }
Oc  void Drive()
Ps  {
q02     static char keycode[] = "\76\55\36\57\100";
5r      register short i, j;
Qi      register struct Spieler *spieler;
DV      register long ri;
rb      struct IntuiMessage *Nachricht;
CD      ULONG Art;
1S      long color = 0, dcolor = 1;
o60 #define DELAY  250
8g  #define DELAY2 500
xB2     short delay = 0;
PLO #ifdef WUNSCH
DI2     for (i=0; i<4; i++) {
Wt4        if (sp[i].crash) delay += DELAY2;
hC2     }
zvO #endif
1F2     while (!ende) {
z34        SetRGB4(vp, 14L, color, 0L, 0L);
```

Listing 1. »Quatron«, ein gut gemachtes, packendes Spiel. Bitte mit dem Checksummer (Seite 68) eingeben. (Fortsetzung)



```
543 hL      SetRGB4(vp, 15L, color>>1, 0L, 0L);
544 1q      color += dcolor;
545 s5      if (color == 0 || color == 15) dcolor = -dcolor;
546 TS      spieler = &sp[0];
547 NS      for (i=0; i<4; i++) {
548 fc6       if (!spieler->crash) {
549 i58         SetBlock(i, spieler->x, spieler->y);
550 c6          if (Nachricht = (struct IntuiMessage *) GetMsg(win->
            UserPort)) {
551 MeA           Art = Nachricht->Class;
552 3B5        key = Nachricht->Code;
553 zx         ReplyMsg(Nachricht);
554 v9         if (Art == CLOSEWINDOW) ende = quit = TRUE;
555 xS8          }
556 bY           if (key == 69) ende = TRUE;
557 wC           ri = spieler->r;
558 pn           if (spieler->mensch) {
559 dC5        if (i==2) {
560 duC              if (j = Keytest(keycode)) {
561 MB9          if (j==5) Sprung(spieler); else ri = --j;
562 4Z7          }
563 5a5        }
564 r7         else {
565 vc7          if (i) myjoy0(); else myjoy1();
566 yVB              if (myjoy.fire) {
567 219            if (spieler->sprung) Sprung(spieler);
568 Af7            }
569 wC           else {
570 j99            spieler->sprung = TRUE;
571 BQ             if (myjoy.dy) ri = 1+myjoy.dy;
572 AJ             if (myjoy.dx) ri = 2+myjoy.dx;
573 Fk7          }
574 G15        }
575 H1         if ((spieler->r^ri) == 2) ri = spieler->r;
576 In8          }
577 4K           else {
578 Nw5        if (!rnd(50) || TestBlock(spieler->x+xr[ri],
579 N9G                   spieler->y+yr[ri]) > 0)
580 I15        {
581 H1C                ri=(ri+rnd(2)*2+1)%4;
582 St                 if (TestBlock(spieler->x+xr[ri],
583 RDB              spieler->y+yr[ri]) > 0)
584 V6E                  ri = (ri+2)%4;
585 RwA              }
586 Sx8          }
587 fd           spieler->x+=xr[ri];
588 ql           spieler->y+=yr[ri];
589 10           spieler->r = ri;
590 Ye3       j = TestBlock(spieler->x,spieler->y);
591 2d        if (j==-1) spieler->count += 10;
592 Um8          Print(40L+i*80L, 203L, 1L, Zahl(++spieler->count, 4
             ));
593 8Q3       if (j > 0) {
594 b1A            spieler->crash = TRUE;
595 Rc5         for (j=2; j<5; j++) {
596 7YC              SetAPen(rp, (long)j);
597 HS7           Move(rp, 8L+i*80L, 203L+j);
598 Vv             Draw(rp, 39L+i*80L, 203L+j);
599 fA5         }
600 Gt          if ((delay += DELAY2) == DELAY2*4) ende = TRUE;
601 hC8           }
602 Gf            else SetBlock(i+6, spieler->x, spieler->y);
603 jE6         }
604 i6          else for (j=0; j<delay; j++);
605 LP          spieler++;
606 mH4       }
607 cD        for (j=0; j<DELAY; j++);
608 oJ2     }
609 pK0 }
610 08  void Sprung(hops)
611 Gz  register struct Spieler *hops;
612 oH  {
613 Mp2   register long zufall;
614 uN    if (hops->count < 25) return;
615 gS    hops->sprung = FALSE;
616 8d    hops->count -= 25;
617 2e    zufall = rnd(3)+2;
618 vu    hops->x += xr[hops->r]*zufall;
619 23    hops->y += yr[hops->r]*zufall;
620 OV0 }
(C) 1987 M&T
```

Listing 1. »Quatron«, ein gut gemachtes, packendes Spiel. Bitte mit dem Checksummer (Seite 68) eingeben. (Schluß)

Programmname:	myjoy
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Assembler
Assembler:	Assembler von Aztec-C V3.4
Aufrufe:	siehe Text

```
Programm : myjoy
----------------

 1 td0 joy0dat      equ   $DFF00A
 2 D5  joy1dat      equ   $DFF00C
 3 AG  joyfire      equ   $BFE001
 4 XX  FIRE0_BIT    equ   6
 5 Jg  FIRE1_BIT    equ   7
 6 Jh  ; -->  Hier sind die Joystick-Str
       ukturen
 7 VC3    dseg
 8 5k     public   _myjoy
 9 YM     bss     _myjoy,10
10 kd0 ; -->  Hier beginnt das 68000-Pro
       gramm
11 XD3    cseg
12 dd     public _myjoy0
13 eY0 _myjoy0:
14 xi3    movem    d4/a6,-(sp)
15 Z4     move.l   #FIRE0_BIT,d4
16 d9     move.w   joy0dat,d0
17 R2     bra.s    joystick
18 Ab0 ;
19 ok3    public   _myjoy1
20 oj0 _myjoy1:
21 4p3    movem    d4/a6,-(sp)
22 mD     move.l   #FIRE1_BIT,d4
23 oL     move.w   joy1dat,d0
24 ji0 ; -->  Hier folgt das joystick-Un
       terprogramm
25 iQ  joystick:
26 1b3    lea     _myjoy,a6
27 8W     clr.l    (a6)
28 CV     clr.l    4(a6)
29 8H     btst     #9,d0
30 SH     beq.s    nichtlinks
31 7N     move.l   #-1,(a6)
32 1x0 nichtlinks:
33 YZ3    btst     #1,d0
34 sB     beq.s    nichtrechts
35 Ou     move.l   #1,(a6)
36 4L0 nichtrechts:
37 VV3    lsr.w    #1,d0
38 Yb     roxl.b   #1,d1
39 XX     lsr.w    #1,d0
40 fJ     roxl.b   #1,d2
41 Uk     eor.b    d1,d2
42 ik     btst     #0,d2
43 S1     beq.s    nichtunten
44 J6     move.l   #1,4(a6)
45 vf0 nichtunten:
46 Eq3    lsr.w    #7,d0
47 hk     roxl.b   #1,d1
48 gg     lsr.w    #1,d0
49 os     roxl.b   #1,d2
50 dt     eor.b    d1,d2
51 rt     btst     #0,d2
52 sl     beq.s    nichtoben
53 Wh     move.l   #-1,4(a6)
54 BT0 nichtoben:
55 1g3    clr.w    8(a6)
56 2Y     btst     d4,joyfire
57 Yr     bne.s    nichtfire
58 ZB     move.w   #1,8(a6)
59 Qk0 nichtfire:
60 8L3    movem    (sp)+,d4/a6
61 34     rts
62 Or     end
(C) 1987 M&T
```

Listing 2. Der Assemblerteil von »Quatron« dient zur schnellen Abfrage der zwei Joystickports

2-Megabyte-Speichererweiterung für AMIGA 500 und AMIGA 1000

Die Karten sind für beide Rechner verwendbar

Version a: 2 Mega Byte bestückt
Version b: 2 Mega Byte bestückt, erweiterbar auf 4 Mega Byte
Erweiterung intern im CPU-Sockel einsteckbar

Einbinden ins Betriebssystem mit addmem
2 bzw. 4 Anfangsadressen einstellbar (ab 200000)
Aufgebaut mit 1-Megabit-Chips
9 Bit Hidden Refresh

GIGATRON ★ 4590 Cloppenburg
G. Preuth, R. Tiedeken
Resthauserstraße 128, Telefon 04471/3070
Dienstag und Donnerstag 15.00–18.30 Uhr

Ecosoft Economy Software AG

Postfach 1905, D 7890 Waldshut, Tel. 077 51 - 79 20
Villa Domingo, CH 6981 Astano, Tel. 091 - 73 28 13

Prüf-Software und Frei-Programme (fast) gratis

Über 3'000 Disketten mit professioneller Prüf-Software (Programme vor Anwender-Registrierung prüfen), aber auch nützlichen Amateur-Programmen für den beruflichen und privaten Gebrauch für IBM-PC/Kompatible, Macintosh, Atari ST, Amiga, C64/128, Apple II.

Katalog auf Disketten und Verzeichnis DM 10.-
(Bitte Computermodell angeben und Banknote oder Scheck beilegen.)

Neu: Emulation von Fremdsoftware, z.B. MS-DOS auf Amiga, C64 auf Amiga, Macintosh auf Atari ST, usw. Gratis Info.-Schrift verlangen. 181

Gegen Einsendung dieses Inserates erhalten Sie einen Gutschein für 1 Gratis-Ecosoft-Diskette.

AMIGA Peripherie
vom Commodore Vertragspartner

Doppellaufwerk	**598DM**
Einzellaufwerk	**349DM**
5.25 " Laufwerk	**458DM**
Einbaulaufwerk für Amiga 2000 mit Anleit. + Zube.	**269DM**
Midi Interface f. Amiga 1000	**98DM**
Midi Interface f. Amiga 500/2000	**98DM**
Soundsampler - Amiga 500/2000	**198DM**
512 KB Ram f. Amiga 500 m.Uhr	**298DM**
256 KB Ram f. Amiga 1000	**149DM**
1 MB Ram für Amiga 1000	**648DM**
512 Kb Ram für A2000 Erweiterung	**179DM**
Deluxe-Music (Midi-fähig)	**198DM**
Transformer + MS-Dos	**170DM**
PC-Karte incl. 5.25 " Laufwerk	**1130DM**
AT-Karte incl. 5.25 " Laufwerk	**1630DM**
Filecard 20 MB f. A2000 + PC Card	**798DM**
AMIGA 500	**1130DM**
AMIGA 2000 m. 2-Diskettenlaufw. u. Monitor 1084	**3245DM**
2 MB Ram für Amiga 2000	**850DM**
Pal Karte-A2000	**210DM**
20 MB Harddisk + SCSI Controler für Amiga 2000	**1400DM**



Helmut Adler
Computer - Technologie
Schlägel & Eisen Straße 9
4352 Herten 3

Telefon 02366/55891

Das AMIGA-Projekt
»DE LUXE SOUND V.2.2. PLUS«
NEU mit RECORDMAKER V.2.2.
DER AUDIODIGITIZER DER LUXUSKLASSE

»Getestet von guten Computer-Fachmagazinen«

AMIGA 12/87 · KICKSTART 12/87 · AMIGA AKTIV 8/87

Hier einige Features von De Luxe Sound Plus in Stichworten:

- Der brandneue »RECORDMAKER« erlaubt jetzt DIRECT-SAMPLING (mit oder ohne Vorspannbild) auf bis zu 255 DISKETTEN NONSTOP, wenn zwei Laufwerke vorhanden sind
- Erzeugen von SOUNDS im STANDARD-FORMAT (DUMP-FORMAT)
- Erzeugen von SOUNDS im IFF-FORMAT
- Erzeugen von IFF-INSTRUMENTS (für z.B.: DE LUXE MUSIC C. SET)
- Erzeugen von SONIX-INSTRUMENTS (Perkussiv) - Pauke etc.
- Erzeugen von SONIX-INSTRUMENT (mit LOOPING) - Trompete etc.
- ECHO- & HALl-EFFEKTE in Stereo mit allen fertigen Soundsamples (SOUNDS im
- STANDARD-FORMAT)
- SAMPLER als ECHO-HALLGERÄT einsetzen (ohne SAMPLING)
- Klangverfremdungen (AM- FM-MODULATIONEN)
- DE LUXE SOUND ist voll FAST-RAM-kompatibel
- Regelbarer Vorverstarker bereits eingebaut
- Superschnell und Superkurz da komplett in ASSEMBLER
- Unsere HARDWARE arbeitet zusätzlich mit fremder STEUERSOFTWARE z.B.: AEGIS AUDIOMASTER, STUDIO MAGIC, FUTURE SOUND ETC.

DE LUXE SOUND PLUS fur AMIGA 1000 komplettes Gerät auschlußfertig mit Steuersoftware, Anleitung, DEMOSOUNDS & RECORDMAKER 2.2. nur DM 198,-
DE LUXE SOUNDS PLUS für AMIGA 500/2000 kompl. Gerät anschlußfertig mit Steuersoftware, Anleitung, DEMOSOUNDS & RECORDMAKER 2.2. nur DM 228,-
DE LUXE SOUND DEMO DISK MIT Originalanleitung & DEMOSOUNDS nur DM 10,-
MIC 600 passendes dynamisches Richtmikrofon mit Ein-Ausschalter und 3 m Anschlußkabel für DE LUXE SOUND SAMPLER nur DM 25,-
AK 2 Adapterkabel 2 m für ältere Stereoanlagen (mit nur DIN-Ausgängen) an unseren SAMPLER (Cincheingang) nur DM 7,-
UPDATE-SERVICE für RECORDMAKER (02381) 67 31 65

hagenau computer
Münsterstraße 202 · 4700 Hamm 5 · Ruf: (02381) 67 31 65

Wir liefern bestmöglich per Nachnahme oder Vorkasse ab Lager Hamm zuzüglich Versandspesen zu Selbstkosten.

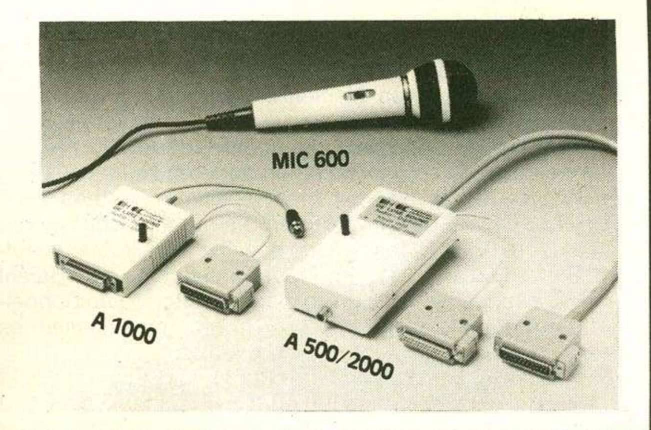
Neu !!! Das AMIGA MIDI-INTERFACE mit Gehäuse NEU !!!
Unser neues AMIGA-MIDI-Interface besitzt alle wichtigen Ein- und Ausgänge wie: 2 x MIDI-IN, 1 x MIDI-OUT, 1 x MIDI-THRU sowie ein Gehäuse und ein Anschlußkabel für den seriellen Port RS 232. Gern liefern wir Ihnen auch ein passendes MIDI-Keyboard von fast allen namhaften Herstellern (z.B. **CASIO, YAMAHA, ROLAND** usw.)
AMIGA-MIDI kostet anschlußfertig mit Gehäuse nur 98,- DM

AMIGA-Diskettenlaufwerke für A 500 - 1000 - 2000
Ausführung in Metallgehäuse, mit Driveabschalter u. Busdurchführung

Einzellaufwerk	3,5 Zoll	(Ausführung s.o.)	nur 369,- DM
Doppellaufwerk	3,5 Zoll	(Ausführung s.o.)	nur 698,- DM
Einzellaufwerk	5,25 Zoll	(Ausführung s.o.)	nur 448,- DM
KOMBI 5,25 +	3,5 Zoll	(Ausführung s.o.)	nur 798,- DM

EASYTITLE Ein superkurzer Titelmaker!! Lädt ILBM-LORES, MEDRES, INTERLACED, HIRES (alle Auflösungen auch in PAL) ferner HAM-BILDER, von DIGIPAINT & PRISM und SOUNDS im STANDARD-FORMAT 29,- DM
BOOT-TITEL erzeugt einen TITELVORSPANN im BOOTSECTOR in 4096 verschiedenen Farben mit diversen DPAINT-BRUSHES 29.- DM
STRING-REPLACER Das Suchen, Ersetzen und Verändern von Texten im ASCII-Format wird zum reinen Vergnügen. Der Original-Text und Ersatz-Text werden gleichzeitig angezeigt 29,- DM
SUPER-MON ist ein komfortabler Speichermonitor mit exclusiven Funktionen und Diskoperationen. Logischer Disassembler mit 68000 / 68010 Mnemonics-Befehlen. Register anzeigen und ändern usw. 49,- DM

THE BEST OF PUBLIC DOMAIN
ES-PD-BOX 1-11 Jede Box enthält 5 ausgesuchte, individuell nach Themengebieten bespielte Disketten mit TOP-PD-Software, z.B. Musik, Grafik, Tools, usw. usw. Preis pro PD-Box 39,90 DM
Fred FISH DISK 1-110 Jede Disk nur lächerliche 4,50 DM
In Vorbereitung: AMIGA PAL VIDEODIGITIZER / Echtzeituhr für AMIGA 1000



Die Maus lebt!

Jeder Programmierer will, daß seine Programme einen besonderen Eindruck hinterlassen. Mit dem Editor von »MouseCreator« lassen sich leicht eigene Mauszeiger erstellen. Aber das Basic-Programm kann noch einiges mehr.

Eigene Mauszeiger in Basic zu verwenden ist nicht weiter schwer, aber zuerst muß man die nötigen Daten besitzen. MouseCreator ist ein komfortabler Editor mit vielen Fähigkeiten, die im folgenden vorgestellt werden. Der Editor ist vollständig mausgesteuert, bis auf die Eingabe der Dateinamen. Außerdem ist noch etwas ganz Besonderes in das Programm eingebaut, doch dazu nach der Bedienung.

Damit das Programm sich starten läßt, brauchen Sie im selben Verzeichnis, in dem das Amiga-Basic steht, noch die Dateien »dos.bmap« und »graphics.bmap«. Aus diesen Libraries (Bibliotheken) werden einige Funktionen benötigt.

Nach dem Start des Programms sehen Sie den Bildschirm des Editors (siehe Bild). Das Feld in der Mitte ist nur ein Ausschnitt der Fläche, in der Sie zeichnen können. Mit MouseCreator können Sie Zeiger bis zu einer Größe von 16 x 107 Punkten erstellen. Um diesen Ausschnitt zu verschieben, verwenden Sie das Gadget (Schieberegler) rechts daneben. Wenn Sie den Regler anklicken und die Taste gedrückt lassen, können Sie ihn beliebig bewegen. Durch Anklicken der Pfeile wird der Ausschnitt immer um eine Zeile nach oben beziehungsweise unten verschoben. Rechts neben diesem Gadget sehen Sie Ihren Mauszeiger in Originalgröße, wodurch Sie auch den Ausschnitt richtig positionieren können.

Wie bei Mauszeigern üblich, können Sie vier Farben verwenden. Die vier Felder links neben dem Zeichenfeld dienen zur Auswahl der Farbe, wobei die aktuelle an einem hellen Rahmen erkennbar ist. Um eine andere Farbe zu wählen, klicken Sie einfach das entsprechende Feld an. Die oberste der vier Farben ist später transparent, das heißt, daß dahinterliegende Farben zu sehen sind. Die anderen drei können nach Belieben mit den drei Schiebereglern unter dem Zeichenfeld verändert werden. Dabei wird immer die aktuelle Farbe eingestellt. Es gibt wieder zwei Wege, die Werte für rot, grün und blau zu verändern. Entweder Sie verschieben den Regler wie bei dem anderen schon beschrieben, oder Sie klicken auf den Feldern rechts und links daneben. Bei Mausklick auf den linken Feldern (R, G, B) vermindern Sie den Wert. Die rechten Felder zeigen den Farbanteil (0 bis 15) an und erhöhen bei Anklicken den Farbwert um 1.

Um nun mit den eingestellten Farben zu zeichnen, bewegen Sie die Maus in die Zeichenfläche und klicken mit der Maus das gewünschte Rechteck an. Ein sehr großer Vorteil von MouseCreator ist, daß man auch mehrere Punkte nacheinander setzen kann, um zum Beispiel Linien zu ziehen. Dazu klickt man ein Rechteck an, läßt die Maustaste gedrückt und bewegen die Maus Ihren Wünschen entsprechend über den Ausschnitt.

Nun noch zu den sechs Schaltern rechts neben den Farbreglern. Die Schalter »Save« und »Load« dienen zum Laden und Speichern der mit MouseCreator generierten Mauszeiger. Sie können also auch später noch alte Zeiger laden, editieren und wiederverwenden. Mit einem kleinen Unterprogramm wird dabei auch festgestellt, ob die Datei existiert. Wenn Sie probieren, einen Zeiger in eine Datei zu speichern, die schon existiert, erscheint eine Fehlermeldung, nach der Sie einen anderen Namen wählen können. Wollen Sie dann nicht mehr speichern, drücken Sie <RETURN>. Beim Speichern dürfen Sie nur einfache Namen verwenden (zum Beispiel maus1), da sonst später Probleme auftauchen. Sie sollten also vor dem Programmstart mit dem Befehl CHDIR das gewünschte Directory festlegen. Durch einen Doppelklick auf den Schalter »Quit« wird das Programm verlassen, es wird aber keine Sicherheitsabfrage durchgeführt.

Um den sogenannten Hotpoint (der Punkt, mit dem angeklickt werden kann) zu setzen, klicken Sie auf den Schalter »Point« und danach auf den gewünschten Punkt im Zeichenfeld. Der Schalter »Clear« kann auf zwei Weisen benutzt werden. Durch einfaches Anklicken wird nur der sichtbare Ausschnitt gelöscht. Mit einem Doppelklick löschen Sie die gesamte Zeichnung, aber hier ist etwas Vorsicht geboten, denn es erfolgt keine Sicherheitsabfrage.

Der letzte Schalter (»Reset«) stellt die Farben ein, die am Anfang des Programms verwendet werden.

Wenn Sie nach dem Zeichnen und Speichern im aktuellen Verzeichnis nachschauen, entdecken Sie zwei neue Dateien. Haben Sie beim Speichern zum Beispiel den Namen »maus1« eingegeben, sehen Sie nun die Dateien »maus1.mouse« und »maus1.sub«. Die erste Datei enthält alle nötigen Zahlenwerte des Zeigers und kann zum Beispiel auch von C-Programmierern verwendet werden. Die zweite Datei ist nur für die Basic-Programmierer interessant, da sie ein Unterprogramm enthält, das den Zeiger erzeugt und für das aktuelle Ausgabefenster von Basic bereit stellt.

Bei dem Einbinden der Routine müssen Sie nur auf eine Kleinigkeit achten. In dem Unterprogramm wird eine Bibliotheksfunktion verwendet, die in der »intuition.library« steht. Sie benötigen also die Datei »intuition.bmap«. Eine der ersten Zeilen Ihres Hauptprogramms muß also so aussehen:

```
LIBRARY "intuition.library"
```

Mit dem Befehl MERGE fügen Sie die Datei nun an Ihr Basic-Programm an. Wenn Sie dann Ihren Zeiger auf dem Bildschirm sehen wollen, rufen Sie die Funktion folgendermaßen auf:

```
CALL maus1
```

Die Routine erhält den Namen, den Sie beim Speichern im Editor angegeben haben. Jetzt sollten Sie Ihren »persönlichen« Mauszeiger schon sehen. Aber es wird noch besser. Erstellen Sie mit MouseCreator einen zweiten Zeiger und laden Sie das zweite Unterprogramm noch in Ihr Hauptprogramm dazu. Wenn Sie jetzt abwechselnd die zwei Routinen aufrufen, ändert sich Ihr Mauszeiger dementsprechend. Dadurch können Sie beispielsweise mit zwei Bildern, die die Texte »Bitte« und »warten« enthalten, dem Benutzer zeigen, daß das Programm noch läuft.

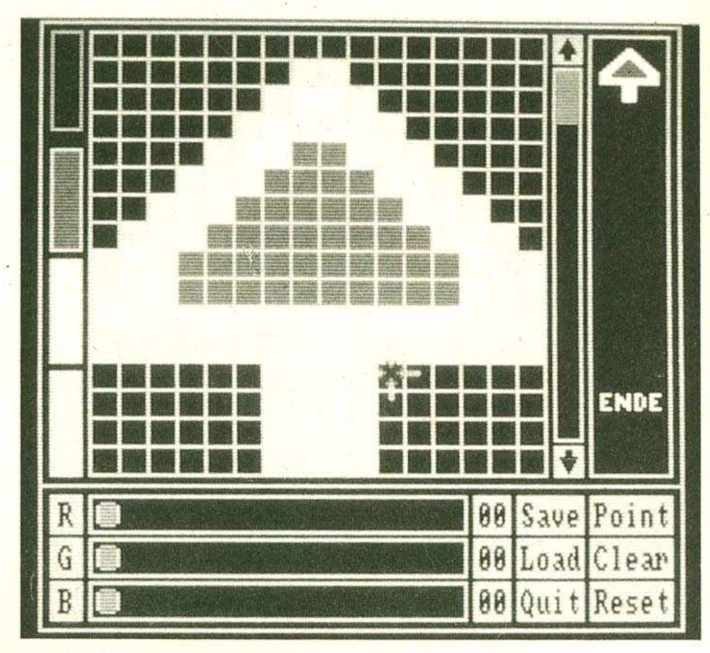


Bild. So sieht der Editor von MouseCreator aus.

Natürlich können Sie auf diese Weise beliebig viele Zeiger in einem Programm verwenden. Die Anzahl ist eigentlich nur dadurch begrenzt, daß das Programm unbedingt ohne Fast-Memory laufen muß und der Speicher dann nicht mehr ausreicht. Die Daten für die Mauszeiger müssen unbedingt im Chip-Memory liegen, da sie ja Grafikdaten darstellen.

Für die Besitzer eines AC/Basic-Compilers gibt es noch eine gute und eine schlechte Nachricht: Das Programm läßt sich fast ohne Änderung übersetzen, aber zwei Kleinigkeiten funktionieren leider nicht. Die Doppelklicks auf »Clear« und »Quit« werden nicht erkannt. Man muß also das Gesamtbild löschen, indem man die einzelnen Teilstücke löscht. Zum Verlassen drückt man nur

<Ctrl c>. Aber vielleicht liefert ja ein Leser eine elegantere Lösung. Beim Compilieren setzen Sie bitte nur die Schalter »N« und »R« auf an.

Unter Berücksichtigung dieser geringen Einschränkungen sind Sie aber mit MouseCreator in der Lage, Ihr Programm aufzuwerten. Die Wirkung ist jedenfalls im Verhältnis zum Aufwand sehr hoch. (Thomas Wimmer/rb)

Programmname:	MouseCreator
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic 1.2

```
STO ' MouseCreater V1.2
Er  ' von: Thomas Wimmer
IT  DECLARE FUNCTION Move& LIBRARY
b5  DECLARE FUNCTION Lock& LIBRARY
nI  LIBRARY ":dos.library"
Go  LIBRARY ":graphics.library"
HK  SCREEN 2,640,200,4,2
OF  WINDOW 1,"MouseCreater V1.2",,16,2
g5  DIM SHARED Col(15,2),Slide1%(75),Slide2%(50),Block&(220)
H1  FOR i1=0 TO 15
2F3    FOR i2=0 TO 2:READ Col(i1,i2):NEXT i2
y9     PALETTE i1,Col(i1,0)/15,Col(i1,1)/15,Col(i1,2)/15
TOO NEXT i1
9b  FOR i1=1 TO 21
ye3    READ x1,y1,x2,y2,z:LINE (x1+140,y1)-(x2+140,y2),z,bf
W30 NEXT i1
8Z  FOR i1=1 TO 11
xL3    READ x1,y1,x2,y2
E1     LINE (x1+140,y1)-(x2+140,y2),6,b:LINE (x1+141,y1)-(x2+139,y2),6,b
a70 NEXT i1
FB  FOR i1=26 TO 266 STEP 15:LINE (i1+140,3)-(i1+141,131),6,b:NEXT i1
kt  FOR i1=3 TO 131 STEP 8:LINE (166,i1)-(407,i1),6:NEXT i1
id  FOR i1=1 TO 14:READ x1,y1,z:LINE (x1+140,y1)-(x1+z+140,y1),0:NEXT i1
Km  FOR i1=1 TO 12
2k3    READ x%,y%,z%,Text$:COLOR 0,z%:x%=x%+140:Display x%,y%,Text$
gDO NEXT i1
nF  LINE (170,140)-(171,144),6,bf:LINE (172,139)-(174,145),8,bf
3z  LINE (175,139)-(176,145),7,bf:LINE (177,139)-(179,145),8,bf
Z9  LINE (180,140)-(181,144),6,bf:GET (413,14)-(424,29),Slide1%
72  GET (170,139)-(181,145),Slide2%:PUT (170,139),Slide2%
Ow  DATA 0,0,0,0,0,0,0,0,0,0,15,0,15,0,15,0,0,0,10,15,0,3,11
Nk  DATA 0,7,12,0,10,15,0,10,15,0,10,15,0,10,15,0,10,15,0,10,15,0,13,13
4W  DATA 6,3,22,32,2,6,38,22,67,3,6,70,22,99,4,6,102,22,131,5,271,3,286,11,9
bS  DATA 271,123,286,131,10,273,14,284,29,7,6,137,22,147,6,6,149,22,159,6,6
h4  DATA 161,22,171,6,228,137,247,147,6,228,149,247,159,6,228,161,247,171,6
e0  DATA 251,137,286,147,11,251,149,286,159,12,251,161,286,171,6,290,137,333
7i  DATA 147,13,290,149,333,159,14,290,161,333,171,6,290,3,333,131,2,26,3
0a  DATA 267,131,2,0,0,339,134,290,3,333,131,271,13,286,121,0,134,339,174,26
ex  DATA 137,224,147,26,149,224,159,26,161,224,171,6,3,22,32,6,38,22,67,6,70
ka  DATA 22,99,6,102,22,131,278,4,1,277,5,3,276,6,5,275,7,7,274,8,9,277,9,3
81  DATA 277,10,3,277,124,3,277,125,3,274,126,9,275,127,7,276,128,5,277,129
4Q  DATA 3,278,130,1,10,145,6,"R",10,157,6,"G",10,169,6,"B",230,145,6,"00"
wI  DATA 230,157,6,"00",230,169,6,"00",253,145,11,"Save",253,157,12,"Load"
Eo  DATA 253,169,6,"Quit",292,145,13,"Point",292,157,14,"Clear",292,169,6
cK  DATA "Reset"
og  COLOR 6,0:LOCATE 1,1
H8  ActCol=2:OldCol=2:Info=0:Flag=0:Scr=0:HPx=0:HPy=0:HP=0
Qd  SetSlide ActCol:SetActCol ActCol,OldCol
F5  ON MOUSE GOSUB CheckMouse:MOUSE ON
v1  WaitForMouse: IF Info=1 THEN GOTO MousePressed ELSE GOTO WaitForMouse
Rx  CheckMouse:   m=MOUSE(0):x=MOUSE(1):y=MOUSE(2):Info=1:RETURN
KN  MousePressed: Info=0:IF Flag=1 THEN RETURN
k0  IF x>166 AND x<407 AND y>3 AND y<131   THEN GOTO SetPoint
TD  IF x>411 AND x<426 AND y>3 AND y<11    THEN GOTO ScrollUp
Fa  IF x>411 AND x<426 AND y>123 AND y<131 THEN GOTO ScrollDown
NX  IF x>167 AND x<363 THEN
up3    IF y>137 AND y<171 THEN Slide=INT((y-137)/12):GOTO SetHoriSlide
s10 END IF
uD  IF x>368 AND x<387 THEN
F13    IF y>137 AND y<171 THEN AddS=INT((y-137)/12):GOTO AddSlide
vo0 END IF
BF  IF x>146 AND x<162 THEN
193    IF y>137 AND y<171 THEN SubS=INT((y-137)/12):GOTO SubSlide
yr0 END IF
EI  IF x>146 AND x<162 THEN
na3    IF y>3 AND y<32    THEN NewAct=2:GOTO NewColor
gk     IF y>38 AND y<131 THEN NewAct=INT((y-38)/32)+3:GOTO NewColor
2v0 END IF
Ra  IF x>391 AND x<426 THEN
lI3    IF y>137 AND y<147 THEN GOTO SavePointer
pm     IF y>149 AND y<159 THEN GOTO LoadPointer
Ye     IF y>161 AND y<171 THEN GOTO QuitMC
700 END IF
FK  IF x>430 AND x<473 THEN
RA3    IF y>137 AND y<147 THEN GOTO SetHotPoint
qD     IF y>149 AND y<159 THEN GOTO ClearPointer
qu     IF y>161 AND y<171 THEN GOTO ResetColors
C50 END IF
Rh  IF x>411 AND x<426 AND y>13 AND y<121 THEN SetVertSlide
5e  GOTO WaitForMouse
Us  ScrollUp:
a5  Scr1=9:Scr2=0:Scr3=-1:Scr4=121:Scr5=8:Scr6=0:Scr7=0:Scr8=6:Scr9=0
f3  GOTO Scrolling
xQ  ScrollDown:
e9  Scr1=10:Scr2=91:Scr3=1:Scr4=0:Scr5=-8:Scr6=121:Scr7=120:Scr8=21:Scr9=128
YH  Scrolling:
bz  PALETTE Scr1,0,13/15,13/15:IF Scr=Scr2 THEN GOTO ScrollEnd
HC  PUT (413,14+Scr),Slide1%:Scr=Scr+Scr3:PUT (413,14+Scr),Slide1%
pa  LINE (166,3+Scr4)-(407,10+Scr4),0,bf:SCROLL (166,3)-(407,131),0,Scr5
Ji  LINE (166,3+Scr6)-(407,10+Scr6),0,bf
HF  FOR i1=166 TO 406 STEP 15:LINE (i1,3+Scr7)-(i1+1,11+Scr7),6,b:NEXT i1
Eb  LINE (166,3+Scr9)-(407,3+Scr9),6:NewLine=Scr+Scr8:Yp=4+Scr7
dZ  Redraw NewLine,Yp:IF MOUSE(0)=-1 THEN GOTO Scrolling
2W  ScrollEnd: PALETTE Scr1,0,10/15,1:GOTO WaitForMouse
5u  NewColor: SetActCol NewAct,ActCol:SetSlide ActCol:GOTO WaitForMouse
7I  SetPoint:
hF  Xp=INT((MOUSE(1)-166)/15):IF Xp<0 THEN Xp=0 ELSE IF Xp>15 THEN Xp=15
gc  Yp=INT((MOUSE(2)-3)/8):IF Yp<0 THEN Yp=0 ELSE IF Yp>15 THEN Yp=15
jv  LINE (Xp*15+168,Yp*8+4)-(Xp*15+180,Yp*8+10),ActCol,bf
11  LINE (Xp*2+436,Yp+6+Scr)-(Xp*2+437,Yp+6+Scr),ActCol
U2  IF HP=1 AND Xp=HPDx AND Yp+Scr=HPDy THEN
rZ3    LINE (Xp*15+172,Yp*8+6)-(Xp*15+176,Yp*8+8),6,bf
bUO END IF
Lv  IF MOUSE(0)=-1 THEN GOTO SetPoint ELSE GOTO WaitForMouse
6Y  AddSlide:
Rs  Col(ActCol,AddS)=Col(ActCol,AddS)+1
5i  IF Col(ActCol,AddS)=16 THEN Col(ActCol,AddS)=0
7x  SetSlide ActCol
Zr  PALETTE ActCol,Col(ActCol,0)/15,Col(ActCol,1)/15,Col(ActCol,2)/15
H8  IF MOUSE(0)=-1 THEN GOTO AddSlide ELSE GOTO WaitForMouse
TS  SubSlide:
Xh  Col(ActCol,SubS)=Col(ActCol,SubS)-1
FF  IF Col(ActCol,SubS)=-1 THEN Col(ActCol,SubS)=15
```

Listing. »MouseCreator« dient zur Generierung von eigenen Mauszeigern in Basic-Programmen

```
D3   SetSlide ActCol
fx   PALETTE ActCol,Col(ActCol,0)/15,Col(ActCol,1)/15,Col(ActCol,
     2)/15
e2   IF MOUSE(0)=-1 THEN GOTO SubSlide ELSE GOTO WaitForMouse
zk   ResetColors:
yS   ColData: DATA 0,0,0,0,0,15,0,15,0,15,0,0
gb   RESTORE ColData
kz   FOR i1=2 TO 5
o13     FOR i2=0 TO 2:READ Col(i1,i2):NEXT i2
kv      PALETTE i1,Col(i1,0)/15,Col(i1,1)/15,Col(i1,2)/15
FmO  NEXT i1
ZN   SetSlide ActCol:GOTO WaitForMouse
WH   QuitMC: FOR i1=1 TO 100:NEXT i1
f1   IF MOUSE(0)<>2 THEN GOTO WaitForMouse
hK   WINDOW CLOSE 1:SCREEN CLOSE 2:LIBRARY CLOSE
oj   END
vW   ClearPointer: State=MOUSE(0)
hF   PALETTE 14,0,13/15,13/15:LINE (166,3)-(407,131),2,bf
1x   FOR i1=26 TO 266 STEP 15:LINE (i1+140,3)-(i1+141,131),6,b:NE
     XT i1
Wf   FOR i1=3 TO 131 STEP 8:LINE (166,i1)-(407,i1),6:NEXT i1
mO   IF State=2 THEN
UB3     LINE (432,4)-(471,130),2,bf
2i      PUT (413,14+Scr),Slide1%:Scr=0:PUT (413,14),Slide1%
69      HPx=0:HPy=0:HP=0:PALETTE 14,0,10/15,1:GOTO WaitForMouse
oXO  ELSE
9J3     LINE (432,Scr+6)-(471,Scr+21),2,bf
B40  END IF
Mq   PALETTE 14,0,10/15,1:GOTO WaitForMouse
QJ   SetHoriSlide:
2s   IF x>Col(ActCol,Slide)*12+169 AND x<Col(ActCol,Slide)*12+1
     82 THEN
iu3     OldX=Col(ActCol,Slide)*12+170
bT      Loop1:
dw      WHILE MOUSE(0)=-1
Y16        IF MOUSE(1)=OldX THEN GOTO Loop1
Kz         IF OldX=350 AND MOUSE(1)>350 THEN GOTO Loop1
Dq         IF OldX=170 AND MOUSE(1)<170 THEN GOTO Loop1
fH         NewX=MOUSE(1)
mP         IF NewX>350 THEN NewX=350
JA         IF NewX<170 THEN NewX=170
ER         PUT (OldX,Slide*12+139),Slide2%:PUT (NewX,Slide*12+139
           ),Slide2%
8c         Col(ActCol,Slide)=INT((NewX-170)/12)
Ia         PALETTE ActCol,Col(ActCol,0)/15,Col(ActCol,1)/15,Col(A
           ctCol,2)/15
Zi         Col$=MID$(STR$(Col(ActCol,Slide)),2):x%=370:y%=145+Sli
           de*12
t8         IF LEN(Col$)=1 THEN Col$="0"+Col$
Js         OldX=NewX:COLOR 0,6:Display x%,y%,Col$
zn3     WEND
wm      SetSlide ActCol
WPO  END IF
Ox   GOTO WaitForMouse
So   SetVertSlide:
dF   IF y>Scr+13 AND y<Scr+30 THEN
zk3     OldY=Scr+14
2q      Loop2:
yH      WHILE MOUSE(0)=-1
4V6        IF MOUSE(2)=OldY THEN GOTO Loop2
W5         IF OldY=14 AND MOUSE(2)<14 THEN GOTO Loop2
PS         IF OldY=105 AND MOUSE(2)>105 THEN GOTO Loop2
8m         NewY=MOUSE(2)
fG         IF NewY>105 THEN NewY=105
ns         IF NewY<14 THEN NewY=14
RO         PUT (413,OldY),Slide1%:PUT (413,NewY),Slide1%:OldY=New
           Y
F33     WEND
GB      IF Scr=NewY-14 THEN GOTO WaitForMouse
YQ      PALETTE 9,0,13/15,13/15:PALETTE 10,0,13/15,13/15
Uu      Scr=NewY-14:MOUSE OFF
2q      FOR Yp=4 TO 124 STEP 8
xw6        NewLine=(Yp-4)/8+6+Scr:Redraw NewLine,Yp
hz3     NEXT Yp
G7      MOUSE ON:PALETTE 9,0,10/15,1:PALETTE 10,0,10/15,1
s10  END IF
kJ   GOTO WaitForMouse
zc   SavePointer:
2M   PALETTE 11,0,13/15,13/15:LINE (168,138)-(362,146),0,bf:COLOR
     6,0
EN   IF HP=0 THEN
PN3     z$="  No HotPoint set...  ":Display 180,145,z$
a7      FOR i1=1 TO 5000:NEXT i1:GOTO EndSave
zsO  END IF
IC   InputRoutine 1
c1   IF FileName$="" THEN GOTO EndSave
tW   File$=FileName$+".MOUSE":CheckFile File$
6j   LINE (168,138)-(362,146),0,bf
AF   IF File=1 THEN
UY3     z$="File already exists...":Display 180,145,z$
bG      FOR i1=1 TO 5000:NEXT i1:GOTO SavePointer
700  END IF
NL   LINE (168,138)-(362,146),0,bf:PackMousePicture
jO   IF Last/2<>INT(Last/2) THEN Last=Last+1
oL   row=(Last-2)/2
yD   OPEN File$ FOR OUTPUT AS #1
4J3     FOR i1=2 TO 5
nL6        FOR i2=0 TO 2:PRINT #1,Col(i1,i2):NEXT i2
Y53     NEXT i1
Uu      PRINT #1,HPx:PRINT #1,HPy:PRINT #1,HP
qK      Size%=INT((24*row+6)/2)+1:DIM Pointer%(Size%)
g6      GET (436,6)-(467,5+row),Pointer%
my      PRINT #1,Size%:Wid%=5+row:PRINT #1,Wid%
oC      FOR i1=0 TO Size%:PRINT #1,Pointer%(i1):NEXT i1
Qr      ERASE Pointer%
dQO  CLOSE #1
fs   File$=FileName$+".SUB"
NQ   DIM dat$(100):dat=9
ML   dat$(0)=" "
1O   dat$(1)="SUB "+FileName$+" STATIC"
7X   dat$(2)="   SHARED MC"+FileName$
xA   dat$(3)="   IF MC"+FileName$+"=0 THEN"
rE   dat$(4)="      MC"+FileName$+"=1"
85   dat$(5)="      wid%=16:lin%="+MID$(STR$(row),2)
1X   dat$(6)="      hpx%="+STR$(HPx)+":hpy%="+STR$(HPy)
hn   dat$(7)=" "
k3   dat$(8)="      DATA 0,0"
ZB   dat$(9)="      DATA "
cl   FOR i1=2 TO Last-1
1Q4     Kon=0
CF      dat$(dat)=dat$(dat)+MID$(STR$(Block&(i1)),2)+","
Sn      IF LEN(dat$(dat))>70 THEN
mz7        dat$(dat)=LEFT$(dat$(dat),LEN(dat$(dat))-1)
NE         dat=dat+1:dat$(dat)="      DATA ":Kon=1
eX4     END IF
zWO  NEXT i1
7c   IF Kon=1 THEN
Pp3     dat$(dat)=dat$(dat)+"0,0"
N60  ELSE
iX3     dat$(dat)=LEFT$(dat$(dat),LEN(dat$(dat))-1):dat=dat+1
iJ      dat$(dat)="      DATA 0,0"
1eO  END IF
09   dat$(dat+1)=" "
1y   dat$(dat+2)="      FOR i1=1 TO "+MID$(STR$(Last+2),2)
H5   dat$(dat+3)="         READ Info&"
1C   dat$(dat+4)="         High%=INT(Info&/256)"
6U   dat$(dat+5)="         Low%=Info&-(High%*256)"
bM   dat$(dat+6)="         MC"+FileName$+"$=MC"+FileName$+"$+"
TG   dat$(dat+6)=dat$(dat+6)+"CHR$(High%)+CHR$(Low%)"
pe   dat$(dat+7)="      NEXT i1"
RS   dat$(dat+8)="   END IF
vY   dat$(dat+9)=" "
dx   z$="17,"+MID$(STR$(Col(3,0)),2)+"/15,"
nd   z$=z$+MID$(STR$(Col(3,1)),2)+"/15,"+MID$(STR$(Col(3,2)),2)+"
     /15"
xY   dat$(dat+10)="   PALETTE "+z$
n9   z$="18,"+MID$(STR$(Col(4,0)),2)+"/15,"
um   z$=z$+MID$(STR$(Col(4,1)),2)+"/15,"+MID$(STR$(Col(4,2)),2)+"
     /15"
2e   dat$(dat+11)="   PALETTE "+z$
xL   z$="19,"+MID$(STR$(Col(5,0)),2)+"/15,"
1v   z$=z$+MID$(STR$(Col(5,1)),2)+"/15,"+MID$(STR$(Col(5,2)),2)+"
     /15"
7k   dat$(dat+12)="   PALETTE "+z$
WL   dat$(dat+13)="   CALL SetPointer (WINDOW(7),SADD(MC"+FileNam
     e$
7k   dat$(dat+13)=dat$(dat+13)+"$),lin%,wid%,hpx%,hpy%)"
V8   dat$(dat+14)=" "
JG   dat$(dat+15)="END SUB"
wB   OPEN File$ FOR OUTPUT AS #1
```

Listing. »MouseCreator«

```
262 Rp3    FOR i1=0 TO dat+15:PRINT #1,dat$(i1):NEXT i1
263 TGO CLOSE #1
264 Lx  ERASE dat$
265 pK  EndSave:
266 77  PALETTE 11,0,10/15,13/15:SetSlide ActCol:GOTO WaitForMouse
267 Lj  LoadPointer:
268 ER  PALETTE 12,0,13/15,13/15:LINE (168,150)-(362,158),0,bf:COLOR
         6,0
269 eZ  InputRoutine 2
270 aJ  IF FileName$="" THEN GOTO EndLoad
271 An  File$=FileName$+".MOUSE":CheckFile File$
272 8i  LINE (168,150)-(362,158),0,bf
273 MQ  IF File=0 THEN
274 DS3    z$="File not found":Display 210,157,z$
275 tJ     FOR i1=1 TO 5000:NEXT i1:GOTO LoadPointer
276 OHO END IF
277 3S  LINE (168,150)-(362,158),0,bf:UnPackMousePicture File$
278 2X  LINE (413,14)-(424,120),0,bf:PUT (413,14),Slide1%:Scr=0:MOUS
        E OFF
279 fT  FOR Yp=4 TO 124 STEP 8
280 aZ3    NewLine=(Yp-4)/8+6+Scr:Redraw NewLine,Yp
281 KcO NEXT Yp
282 du  MOUSE ON
283 q0  IF HP=1 THEN
284 ee3    IF ABS(HPy)>15 THEN GOTO EndLoad
285 rW     LINE (ABS(HPx)*15+172,ABS(HPy)*8+6)-(ABS(HPx)*15+176,ABS(
           HPy)*8+8),6,bf
286 YRO END IF
287 FN  EndLoad:
288 ZV  PALETTE 12,0,10/15,13/15:SetSlide ActCol:GOTO WaitForMouse
289 Td  SetHotPoint:
290 te  PALETTE 13,0,13/15,13/15:Flag=1:GOSUB WaitForMouse:Flag=0
291 1f  Xp=INT((x-166)/15):IF Xp<0 THEN Xp=0 ELSE IF Xp>15 THEN Xp
        =15
292 qQ  Yp=INT((y-3)/8):IF Yp<0 THEN Yp=0 ELSE IF Yp>15 THEN Yp=15
293 we  LINE (Xp*15+172,Yp*8+6)-(Xp*15+176,Yp*8+8),6,bf
294 1B  IF HP=1 THEN
295 Ty3    IF ABS(HPy)>Scr-1 AND ABS(HPy)<Scr+16 THEN
296 J06       Col=POINT(ABS(HPx)*15+168,(ABS(HPy)-Scr)*8+4):x=ABS(HP
              x)
297 tr        y=ABS(HPy)-Scr:LINE (x*15+168,y*8+4)-(x*15+180,y*8+10)
              ,Col,bf
298 kd3    END IF
299 le0 END IF
300 ci  HPDx=Xp:HPDy=Yp+Scr:HPx=-Xp:HPy=-Yp-Scr:HP=1
301 6j  PALETTE 13,0,10/15,13/15:GOTO WaitForMouse
302 ey  ' -------------------------------------------------------------
        -------------
303 eK  SUB Display(x%,y%,z$) STATIC
304 zc3    e&=Move&(WINDOW(8),x%,y%):CALL Text(WINDOW(8),SADD(z$),LE
          N(z$))
305 xz0 END SUB
306 HT  SUB SetSlide(ActCol) STATIC
307 Rb3    FOR i1=0 TO 2
308 gH6       LINE (168,138+i1*12)-(362,146+i1*12),0,bf
309 X2        Xp=Col(ActCol,i1)*12+170:Yp=139+i1*12:PUT (Xp,Yp),Slid
              e2%
310 ku        z$=MID$(STR$(Col(ActCol,i1)),2):IF LEN(z$)=1 THEN z$="
              0"+z$
311 Rd        x%=370:y%=i1*12+145:COLOR 0,6:Display x%,y%,z$
312 Ip3    NEXT i1
313 570 END SUB
314 EU  SUB SetActCol(NewCol,OldCol) STATIC
315 5L4    SHARED ActCol
316 ep     FOR i1=1 TO 2
317 3k7       IF i1=1 THEN
318 J1A          Col1=OldCol:Col2=6
319 kT7       ELSE
320 ZiA          Col1=NewCol:Col2=15
321 707       END IF
322 nF        IF Col1=2 THEN Yp=3 ELSE Yp=(Col1-2)*32+6
323 2T        LINE (146,Yp)-(162,Yp+29),Col2,b
324 4V        LINE (147,Yp)-(161,Yp+29),Col2,b
325 V24    NEXT i1
326 7Y     ActCol=NewCol
327 JLO END SUB
328 UZ  SUB Redraw(NewLine,Yp) STATIC
329 8Z4    SHARED HPx,HPy,HP
330 Ox     FOR i1=436 TO 466 STEP 2
331 yn8       Col=POINT(i1,NewLine):Xp=(i1-436)/2*15+168
332 7v        LINE (Xp,Yp)-(Xp+12,Yp+6),Col,bf
333 Yw        IF HP=1 AND ABS(HPx)=(i1-436)/2 THEN
334 P7B          IF ABS(HPy)=NewLine-6 THEN
335 4xE             LINE (Xp+4,Yp+2)-(Xp+8,Yp+4),6,bf
336 MFB          END IF
337 NG8       END IF
338 iF4    NEXT i1
339 VXO END SUB
340 OL  SUB CheckFile(File$) STATIC
341 uu3    SHARED File
342 J4     File$=File$+CHR$(0):Arg%=-2
343 ar     Llock&=Lock&(SADD(File$),Arg%)
344 3v     IF Llock&=0 THEN File=0 ELSE File=1
345 BB     CALL UnLock(Llock&)
346 ce0 END SUB
347 EQ  SUB InputRoutine(Place%) STATIC
348 S63    SHARED FileName$
349 sh     FileName$=""
350 ns     LINE (168,Place%*12+126)-(362,Place%*12+134),0,bf
351 lm     CharX%=194:CharY%=Place%*12+133:z$="-> "+STRING$(20,"_")
352 2F     COLOR 6,0:Display 170,CharY%,z$
353 7U     InkeyLoop:
354 sn     t$=INKEY$:IF t$="" THEN GOTO InkeyLoop
355 ZH     IF t$=CHR$(13) THEN GOTO InkeyEnd
356 mf     IF t$=CHR$(8) THEN
357 6c6       IF CharX%=194 THEN GOTO InkeyLoop
358 cr        FileName$=LEFT$(FileName$,LEN(FileName$)-1)
359 AA        z$=FileName$+"_":Display 194,CharY%,z$
360 4Q        CharX%=CharX%-8:GOTO InkeyLoop
361 le3    END IF
362 Y8     Display CharX%,CharY%,t$
363 BH     FileName$=FileName$+t$
364 cM     CharX%=CharX%+8:IF CharX%=354 THEN GOTO InkeyEnd
365 TV     GOTO InkeyLoop
366 TL     InkeyEnd:
367 xz0 END SUB
368 7q  SUB PackMousePicture STATIC
369 G13    SHARED Last
369 G13    SHARED Last
370 oL     Block=2:Last=3:z1$="in progress":z2$="            ":Num=1
371 cv     FOR i1=0 TO 220:Block&(i1)=0:NEXT i1
372 vH     FOR i1=6 TO 112
373 Uh6       FOR i2=15 TO 0 STEP -1
374 2G9          Col=POINT((15-i2)*2+436,i1)
375 OF           IF Col=3 OR Col=5 THEN Block&(Block)=Block&(Block)+
                 2^i2
376 nj           IF Col=4 OR Col=5 THEN Block&(Block+1)=Block&(Block
                 +1)+2^i2
377 Ow6       NEXT i2
378 FR        IF Block&(Block)<>0 OR Block&(Block+1)<>0 THEN Las
              t=Block+1
379 2g        Block=Block+2:Num=Num+1
380 CV        IF Num/2=INT(Num/2) THEN z$=z1$ ELSE z$=z2$
381 V2        Display 222,145,z$382 Qx3    NEXT i1
383 DFO END SUB
384 wC  SUB UnPackMousePicture(File$) STATIC
385 2T3    SHARED HPx,HPy,HP
386 YF     LINE (432,4)-(471,130),2,bf
387 NP     OPEN File$ FOR INPUT AS #1
388 4J6       FOR i1=2 TO 5
389 jV9          INPUT #1,Col(i1,0),Col(i1,1),Col(i1,2)
390 4F           PALETTE i1,Col(i1,0)/15,Col(i1,1)/15,Col(i1,2)/15
391 Z66       NEXT i1
392 9G        INPUT #1,HPx,HPy,HP:INPUT #1,Size%,Wid%
393 TO        DIM Pointer%(Size%)
394 OL        LINE (436,6)-(467,Wid%),0,bf:GET (436,6)-(467,Wid%),Po
              inter%
395 QB        FOR i1=0 TO Size%:INPUT #1,Pointer%(i1):NEXT i1
396 cP3    CLOSE #1
397 ZP     PUT (436,6),Pointer%:ERASE Pointer%
398 SUO END SUB
(C) 1987 M&T
```

Listing. »MouseCreator« dient zur Generierung von eigenen Mauszeigern in Basic-Programmen. Bitte mit dem Checksummer (Seite 68) eingeben.

Nie wieder Tippfehler!

Endlich ist er da, der Checksummer für den Amiga! Die Zeiten der endlosen Fehlersuche sind damit vorbei. »Checkie42« funktioniert für alle Programmiersprachen von Basic über C bis zum Assembler, wenn sie im ASCII-Format vorliegen. Er ist also universell einsetzbar und genügt allen Ansprüchen.

Damit wirklich jeder Leser unseren neuen Checksummer benutzen kann, wurde er in Amiga-Basic geschrieben. Trotz der sehr hohen Zuverlässigkeit und der strukturierten Programmierung konnte er recht kurz gehalten werden. Dabei bietet »Checkie42« aber allen nur erdenklichen Komfort und Sicherheit.

Nach dem Abtippen (siehe Listing), bei dem Sie diesmal noch gut aufpassen müssen, haben Sie endlich die erwünschte Hilfe beim Abtippen unserer Programme. Ab dieser Ausgabe werden alle Listings mit den Checksummen abgedruckt. Bevor wir zur Bedienung des Programms übergehen, soll zuvor noch erläutert werden, aus welchen Teilen sich eine Zeile in Zukunft zusammensetzt. Ein Beispiel für eine solche Zeile sieht so aus:

```
1 TTO print "Hallo!"
```

Die erste Zahl (»1«) stellt die Zeilennummer dar. Die Werte hierfür reichen von 1 bis maximal 9999. Nach einer Leerstelle folgt dann die eigentliche Prüfziffer, die sich noch mal unterteilt. Die ersten zwei Zeichen sind die Prüfziffern für den Text und die Zeilennummer. Erlaubt sind hier die Zeichen von »0« bis »9«, die großen (»A« bis »Z«) und die kleinen Buchstaben (»a« bis »z«). Durch die große Anzahl von verschiedenen Zeichen wurde erreicht, daß eine zweistellige Prüfsumme vollkommen ausreicht. Um dieselbe Fehlersicherheit mit Ziffern allein zu erreichen, würde man mindestens fünf bis sechs Stellen benötigen.

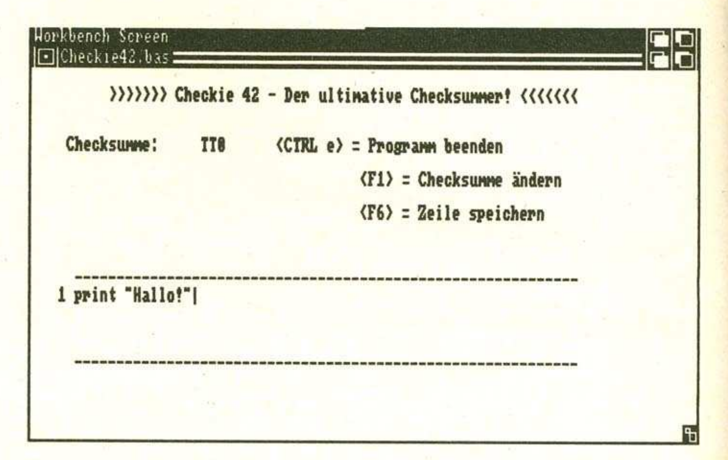
Das letzte Zeichen der Checksumme ist interessant, wenn die abgedruckten Listings strukturiert aufgebaut sind. Die dritte Stelle tritt nur dann auf, wenn sich der Beginn der Zeile verschoben hat. Deshalb erscheint sie auch nicht immer. Nach einer weiteren Leerstelle folgt dann endlich der eigentliche Text.

Doch nun zur Bedienung von »Checkie42«. Nachdem das Programm gestartet wurde, sehen Sie ein Fenster auf dem Bildschirm. In diesem Fenster erscheint zunächst eine Abfrage nach der gewünschten Datei. Hier können Sie einen neuen Dateinamen oder den Namen einer bereits bestehenden Datei eingeben. Zu letzterem Punkt kommen wir später.

Nach der Eingabe, die mit <RETURN> abgeschlossen wird, kommt nun das eigentliche Arbeitsfenster von »Checkie42« (siehe Bild). Links oben blinkt der Text »Checksumme:« und fordert Sie zur Eingabe der im Heft abgedruckten Prüfzeichen auf. Wenn Sie Wert darauf legen, daß das Listing auch bei Ihnen mit den richtigen Einrückungen generiert wird, müssen Sie auch die eventuell vorhandene dritte Stelle mit eingeben. Ansonsten geben Sie nur die ersten zwei Zeichen ein und drücken dann <RETURN>. Ist die dritte Stelle nicht vorhanden, beenden Sie die Eingabe auch mit <RETURN>.

Der Cursor steht nun zwischen den zwei Trennstrichen in der Mitte des Bildschirms. Davor hat das Programm automatisch die Zeilennummer geschrieben. Außerdem wurde der Cursor schon richtig positioniert, wenn die dritte Prüfziffer mit eingegeben wurde. Zum Beenden drücken Sie einfach die RETURN-Taste. Bei fehlerfreier Zeile wird der Raum zum Editieren gelöscht, die nächste Zeilennummer wird angezeigt und das Programm erwartet die Eingabe der nächsten Checksumme.

Leider wird es auch vorkommen, daß der Text einen Fehler enthält. Wenn Sie diesen entdeckt haben, können Sie den Cursor, der ganz am Anfang des Textes steht, an die entsprechende Stelle bewegen. Dabei verwenden Sie die Cursor-Steuertasten, um die Position zu erreichen. Mit der Backspace-Taste löschen Sie dann das Zeichen rechts vom Cursor. Auch das Einfügen von Zeichen ist ohne weiteres möglich. Dazu drücken Sie <TAB>, worauf an der Cursor-Position eine Leerstelle eingefügt wird. Dort können Sie nun das fehlende Zeichen eintragen. Für mehrere Zeichen muß entsprechend oft <TAB> gedrückt werden.



Das übersichtliche Fenster von »Checkie42«, in dem alle Eingaben gemacht und editiert werden können

Wenn Sie die Zeile trotz falscher Checksumme übernehmen wollen, drücken Sie die Taste <F6>. Dies ist zum Beispiel sinnvoll, wenn Sie Kommentare nicht mit eingeben wollen. Natürlich kann es auch vorkommen, daß Sie zwar die Zeile richtig, aber die Prüfsumme falsch eingegeben haben. In diesem Fall können Sie jederzeit die Taste <F1> drücken, worauf Sie die Checksumme ändern können.

Checkie42 speichert die einzelnen Zeilen nach Drücken von <RETURN> oder <F6> sofort in die Datei. Am Schluß des Listings müssen Sie dem Programm aber noch mitteilen, daß das Ende erreicht wurde. Das erreichen Sie durch Drücken von <CTRL E>. Wenn Sie Zeilen trotz falscher Prüfsumme übernommen haben, erfolgt eine Anzeige, wie viele Zeilen es waren. Längere Listings können Sie auch in Etappen abtippen. Wenn Sie keine Lust oder Zeit mehr haben, beenden Sie das Programm mit <CTRL E>. Bei dem nächsten Start geben Sie dann den gleichen Dateinamen ein und Checkie42 sucht automatisch diese Datei. Wenn sie gefunden wurde, lädt das Programm alle Zeilen und ermittelt so die nächste Zeilennummer. Ab dieser Position können Sie dann den Quelltext weiter eingeben.

Noch eine interessante Einzelheit von Checkie42 ist vielleicht für Sie von Interesse: Leerstellen vor und hinter dem eigentlichen Text werden nicht mitgeprüft. Wollten Sie also die Einrückung einer Zeile verändern, so stellt dies kein Problem dar.

Für diejenigen, denen die Basic-Version zu langsam sein sollte, noch ein kleiner Tip. Die Berechnung der eigentlichen Prüfsumme geschieht ab dem Label »CalcSumme:«. Dieser Teil ist sehr leicht in andere Programmiersprachen wie zum Beispiel C umsetzbar.

Nach dem Abtippen sollten Sie Checkie42 testen, indem Sie eines der in dieser Ausgabe abgedruckten Listings abtippen. Probieren Sie jede Funktion des Programmes durch, um etwaige Fehler durch das Abtippen zu entdecken. Wenn Ihr Checksummer dann richtig läuft, brauchen Sie in Zukunft keine Angst mehr vor Tippfehlern zu haben. Sie sparen damit viel Zeit für die vielleicht vergebliche Suche von tückischen Fehlern.

(Dieter Behlich/pa/rb)

Programmname:	Checkie42
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic 1.2

Programm : Checkie42

```
REM ***********************************
REM *** Checksummer: Checkie 42     ***
REM ***        Version 1.0          ***
REM ***********************************
Start:
  GOSUB Init
  GOSUB OpenDatei
  GOSUB Bild
NeueZeile:
  GOSUB Loeschen
  GOSUB EingabeSumme
  IF FEnde=wahr THEN Ende
Wiederholung:
  GOSUB EingabeZeile
  IF FEnde=wahr THEN Ende
  GOSUB CalcSumme
  IF FSumme=falsch THEN Wiederholung
  GOSUB Uebernahme
  GOTO NeueZeile
Ende:
  GOSUB fertig
  END

Init:
  wahr=-1
  falsch=0
  LZeile=240  : REM Anzahl Zeichen/Zeile
  LBZeile=60  : REM Anzahl Zeichen/Bildschrimzeile
  AnzBZeilen=LZeile/LBZeile
  zx=6 : zy=14    : REM Position Zeile
  sx=20 : sy=5    : REM Position Checksumme
  AnzCsZ=3        : REM Anzahl Ziffern/Checksumme
  DIM z(LZeile)
  DIM cs(AnzCsZ)
  cs(AnzCsZ)=0    : REM Zeilenstart
  a=0 : b=0 : c=0  : REM Hilfsvariablen
  i=0 : j=0 : k=0  : REM Zählvariablen
  READ Faktor(i)
  WHILE Faktor(i)<>0
    i=i+1
    READ Faktor(i)  : REM Faktorenreihe
  WEND
  AnzFak=i
  DATA 2,3,4,5,6,0
  RETURN

OpenDatei:
  ON ERROR GOTO Dateifehler
  CLS
  LOCATE 3,5
  dn$="":INPUT "Dateiname: ";dn$
  Zeile=1
  OPEN dn$ FOR INPUT AS #1
  IF FFehler=53 THEN NeueDatei
  WHILE NOT EOF(1)
    LINE INPUT#1,e$
REM PRINT e$
    Zeile=Zeile+1
  WEND
  CLOSE 1
NeueDatei:
  OPEN dn$ FOR APPEND AS #1
  RETURN
Dateifehler:
  FFehler=ERR
  IF FFehler=53 THEN RESUME NEXT
  ON ERROR GOTO 0

Bild:
  CLS
  LOCATE 2,10
  PRINT ">>>>>>> Checkie 42 - Der ultimative Checksummer!
  <<<<<<<"
  LOCATE 13,zx
  FOR i=1 TO 2
    FOR j=1 TO LBZeile
      PRINT "-";
    NEXT j
    LOCATE 13+AnzBZeilen+1,zx
  NEXT i
  LOCATE 5,30
  PRINT "<CTRL e> = Programm beenden"
  LOCATE 7,40
  PRINT "<F1> = Checksumme ändern"
  LOCATE 9,40
  PRINT "<F6> = Zeile speichern"
  RETURN
```

Listing. Der ASCII-Checksummer »Checkie42« verhindert Fehler beim Abtippen von Listings. Bitte ohne Zeilennummern eingeben.

```
Loeschen:
  LOCATE zy,1
  FOR i=1 TO AnzBZeilen
    PRINT TAB(zx);SPACE$(LBZeile)
  NEXT i
  FOR i=0 TO LZeile
    z(i)=32
  NEXT i
  LOCATE zy,1
  PRINT USING "####";Zeile
  aPos=0
  RETURN

EingabeSumme:
  Farbe=0 : i=1
  LOCATE sy,sx+1
  PRINT "..."
blinken:
  Farbe=Farbe XOR 1
  COLOR Farbe,0
  LOCATE sy,sx-15
  PRINT "Checksumme:"
  e$=INKEY$
  IF e$="" THEN blinken
  e=ASC(e$)-48
  IF e=-43 THEN
    FEnde=wahr   : REM <CTRL e>
    GOTO ESEnde
  ELSE
    IF e<>-35 THEN keinEnde   : REM <CR>
    IF i<AnzCsZ THEN blinken
    GOTO ESEnde
  keinEnde:
    IF e<0 THEN blinken
    IF e>9 THEN e=e-7
    IF e<0 THEN blinken
    IF e>35 THEN e=e-6
    IF e<0 THEN blinken
    IF e>61 THEN blinken
    LOCATE sy,sx+i
    COLOR 1,0
    PRINT e$;
    cs(i)=e
    i=i+1
    IF i<=AnzCsZ THEN blinken
  END IF
ESEnde:
  COLOR 1,0
  LOCATE sy,sx-15
  PRINT "Checksumme:"
  RETURN

EingabeZeile:
  x=cs(AnzCsZ)
weiter:
  LOCATE zy+INT(x/LBZeile),zx+(x MOD LBZeile)
  IF x>aPos THEN aPos=x
  e$=INPUT$(1)
  e=ASC(e$)
  IF (e AND 127)<32 THEN Controlcode
  PRINT e$
  z(x)=e : e=30
Controlcode:
  IF e=13 THEN
    RETURN
  ELSEIF e=30 THEN
    a=1
  ELSEIF e=29 THEN
    a=LBZeile
  ELSEIF e=31 THEN
    a=-1
  ELSEIF e=28 THEN
    a=-LBZeile
  ELSE
    GOTO noCrs
  END IF
  x=x+a
  IF x>=0 AND x<LZeile THEN weiter
  x=x-a
  GOTO weiter
noCrs:
  IF e=8 THEN
    FOR i=x TO aPos
      z(i)=z(i+1)
      PRINT CHR$(z(i));
     IF i MOD LBZeile=59 THEN PRINT:PRINT TAB(zx);
    NEXT i
    z(aPos)=32 : PRINT " "
    aPos=aPos-1
  ELSEIF e=9 THEN
    IF aPos>x THEN
      FOR i=aPos TO x STEP -1
        z(i+1)=z(i)
      NEXT i
      z(x)=32
      aPos=aPos+1
      IF aPos=LZeile THEN aPos=aPos-1:z(LZeile)=32
      FOR i=x TO aPos
        PRINT CHR$(z(i));
        IF i MOD LBZeile=59 THEN PRINT:PRINT TAB(zx);
      NEXT i
     END IF
  ELSEIF e=129 THEN
    GOSUB EingabeSumme
    x=cs(AnzCsZ)
  ELSEIF e=134  THEN
    RETURN
  ELSEIF e=5 THEN
    FEnde=wahr
    RETURN
  END IF
  GOTO weiter

CalcSumme:
  a=0 : b=0 : c=0
  IF e=134 THEN
    FSumme=wahr
    FF6=FF6+1
  ELSE
    WHILE z(aPos)=32 AND aPos>0
      aPos=aPos-1
    WEND
    IF aPos>0 THEN
      WHILE z(c)=32
        c=c+1
      WEND
    END IF
    FOR i=c TO aPos
      j=(i-c) MOD AnzFak
      k=(i+1-c) MOD AnzFak
      a=a+((z(i) AND 127)-32)*Faktor(j)
      b=b+((z(i) AND 127)-32)*Faktor(k)
    NEXT i
    FSumme=(cs(1)=(a+Zeile) MOD 62) AND (cs(2)=(b+Zeile) MOD 62)
  END IF
  RETURN

Uebernahme:
  FOR i=0 TO aPos
    PRINT#1,CHR$(z(i));
  NEXT i
    PRINT#1,""
  Zeile=Zeile+1
  RETURN

fertig:
  CLOSE 1
  CLS
  LOCATE 12,35
  PRINT "F E R T I G !!!"
  LOCATE 20,1
  IF FF6<>0 THEN
    PRINT "ACHTUNG!!!  ";
    PRINT FF6;" Zeile(n) wurde(n) ungeprüft gespeichert."
  END IF
  RETURN
(C) 1987 M&T
```

Listing. Der ASCII-Checksummer »Checkie 42« verhindert Fehler beim Abtippen von Listings (Schluß)

DAS AMIGA-GLOSSAR TEIL 6

Die einzelnen Teile des Glossars zum Heraustrennen bilden ein umfassendes Nachschlagewerk der gängigsten Begriffe rund um den Amiga. Sammeln Sie mit.

Sind Sie schon einmal auf den Begriff »treshold« beim Experimentieren mit dem Programm »Preferences« gestoßen? Unter diesem Begriff kann man sich auch mit guten Englisch-Kenntnissen nur wenig vorstellen. Trotz dem mittlerweile guten Angebot an deutschsprachiger Literatur ist die Fachsprache der Computerwelt stark von anglo-amerikanischen Begriffen durchsetzt. Im Amiga-Glossar finden Sie sämtliche Ausführungen nach englischen Stichworten alphabetisch geordnet. Nutzen Sie es als Ergänzung Ihrer Systemdokumentation.

(David Twigg-Flesner/pa)

Notentabelle Das Amiga-Basic-Handbuch enthält zwar eine Notentabelle, aber es fehlen sämtliche Zwischentöne. Die folgende Tabelle enthält die volle Bandbreite, geordnet nach Frequenz und mit Angabe der Oktave und der Nummer für MIDI-Anwendungen (siehe equal tempered scale).

OKTAVE/ Tonlage	Frequenz (Hz)	Midi-Nr.
A	27.50000	21
A#	29.13523	22
H	30.86770	23
C	32.70319	24
C#	34.64782	25
D	36.70809	26
D#	38.89087	27
E	41.20344	28
F	43.65352	29
F#	46.24930	30
G	48.99942	31
G#	51.91308	32
	55.00000	33
	58.27047	34
	61.73541	35
C2	65.40639	36
	69.29565	37
	73.41619	38
	77.78174	39
	82.40688	40
	87.30705	41
Baß	92.49860	42
	97.99885	43
	103.82617	44
	110.00000	45
	116.54094	46
Bariton	123.47082	47
C3	130.81278	48
Tenor	138.59131	49
	146.83238	50
	155.56349	51
	164.81377	52
	174.61411	53
Alt	184.99721	54
	195.99771	55
	207.65234	56

OKTAVE/ Tonlage	Frequenz (Hz)	Midi-Nr.
	220.00000	57
Mezzo-	233.08188	58
Sopran	246.94165	59
C4	261.62556	60
Sopran	277.18263	61
	293.66476	62
	311.12698	63
	329.62755	64
	349.22823	65
	369.99442	66
	391.99543	67
	415.30469	68
Kammer-	440.00000	69
ton	466.16376	70
	493.88330	71
C5	523.25113	72
	554.36526	73
	587.32953	74
	622.25396	75
	659.25511	76
	698.45646	77
	739.98884	78
	783.99087	79
	830.60939	80
	880.00000	81
	932.32752	82
	987.76660	83
C6	1046.50226	84
	1108.73052	85
	1174.65907	86
	1244.50793	87
	1318.51022	88
	1396.91292	89
	1479.97769	90
	1567.98174	91
	1661.21879	92
	1760.00000	93
	1864.65504	94
	1975.53320	95
C7	2093.00452	96
	2217.46104	97
	2349.31814	98
	2489.01586	99
	2637.02045	100
	2793.82585	101
	2959.95538	102
	3135.96348	103
	3322.43758	104
	3520.00000	105
	3729.31009	106
	3951.06641	107
C8	4186.00904	108
	4434.92209	109
	4698.63628	110
	4978.03173	111
	5274.04091	112
	5587.65170	113
	5919.91076	114
	6271.92697	115
	6644.87516	116
ab hier	7040.00000	117
Distor-	7458.62018	118
tion	7902.13282	119

OKTAVE/ Tonlage	Frequenz (Hz)	Midi-Nr.
C9	8372.01808	120
	8869.84419	121
	9397.27257	122
	9956.06347	123
	10548.08182	124
	11175.30340	125
	11839.82152	126
	12543.85395	127
	13289.75032	128
	14080.00000	129
	14917.24036	130

NuDos.library die Bezeichnung der -> dos.library im Programmcode vieler Systemroutinen unter der neuen Version 1.2.

Nyquist sampling (Regeln für Sampling) Beim Amiga tritt der Begriff »sampling« meistens in Verbindung mit dem Digitalisieren von Tönen (Audiodigitalisierung) auf. Ein Signal wird in bestimmten Zeitabständen (hold phase) gemessen und die ermittelten Meßwerte geben in binärer Form den Verlauf des analogen Signals wieder. Gemessen wird für den Amiga die Amplitude (Lautstärke) des Signals als Funktion der Samplingfrequenz. Nach den Nyquist-Regeln kann man die optimale Häufigkeit des Samplings bestimmen, um sicherzugehen, daß a) keine wesentlichen Werte des Eingangssignals verlorengehen und b) die statistische Unabhängigkeit der einzelnen Messungen gewährleistet wird. Grob formuliert besagen die Nyquist-Regeln, daß die Samplingfrequenz doppelt so hoch wie die höchste zu messende Frequenz des Eingangssignals sein sollte, um die beiden genannten Kriterien zu erfüllen (siehe digitizer, quantization).

object code Ein Assembler erzeugt Code in der für den Prozessor notwendigen Maschinensprache. Solcher Code ist nicht unbedingt lauffähig, weil er noch nicht aufgelöste, externe Querverweise enthalten kann. Ein solcher Querverweis ist etwa der Aufruf einer Routine, die sich nicht in der Quelldatei befindet. Solche werden mit Hilfe eines -> link editor eingebunden, um ein -> load file zu erzeugen.

object file Eine Datei, die -> object code enthält.

ObjEd (Objekteditor) Ein Hilfsprogramm für Amiga-Basic, mit dem man die Gestalt eines Sprites beziehungsweise BOBs definieren kann. Bei der Anwendung des Programms ist darauf zu achten, daß die hohe Bildschirmauflösung (640 x 200) benutzt wird. Da Sprites aber stets in niedriger Auflösung dargestellt werden, erscheinen sie beim Editieren nur in halber Breite.

object module disassembler (OMD) Ein Hilfsprogramm bei C-Compilern, welches eine binäre Objektcodedatei in Assembler-Quellcode zurückverwandelt. Unter Umständen werden auch die Quellcodetexte eingefügt (siehe Handbuch zu Lattice C).

offset Der Abstand eines bestimmten Datenwertes (Bytes) von einer vorgegebenen Basisadresse.

Die meisten Systemlibraries haben zum Beispiel ein Offset (auch »bias«) von -30 (dezimal) zwischen Anfangsadresse der library und dem ersten Eintrag in der Sprungtabelle für die in der library enthaltenen Routinen, weil sich zwischen der Basisadresse und dem Beginn der Sprungtabelle noch ein von der library genutzter Datenbereich befindet.

operating system Das Betriebssystem eines Computers. Beim Amiga besteht das »OS« aus mehreren Ebenen. Je höher die Ebene, desto einfacher die Bedienung für den Benutzer. Die einfache Bedienung macht die Programmierung allerdings komplizierter.

Die tiefste Systemebene bildet -> exec. Darüber liegt -> Amiga-DOS, auf das der Anwender über ein -> CLI Zugriff hat. Auf beide Ebenen

baut -> Intuition auf. Die obere Ebene bilden die Anwenderprogramme unter der -> Workbench. Die Schnittstellen, an denen der Benutzer mit dem Computer in Kontakt treten kann, nennt man »user interface« (CLI) beziehungsweise -> iconic interface (Benutzeroberfläche: Workbench).

origin (Ursprungspunkt eines Koordinatensystems) Anders als bei der Mathematik liegt bei einem Bildschirm dieser Punkt in der oberen linken Ecke des Bildes. Dort beginnt der Elektronenstrahl mit dem Abbilden der Grafik. Beim Amiga ist darauf zu achten, daß dieser Punkt nicht immer die (Elektronenstrahl-)Koordinaten 0,0 hat, weil bei der Angabe dieser Koordinate der -> overscan berücksichtigt werden muß. Die Standardwerte für den Beginn der Bildschirmdarstellung sind die Koordinaten 64,44 (x,y). Bei der Positionierung von Sprites (insbesonders -> simple sprites) muß das beachtet werden. Ein Sprite erscheint sonst unter Umständen nicht auf dem Bildschirm.

overlay (Überlagerungsmodul) Ein Programm oder Codemodul, das während des Programmverlaufs nachgeladen wird und einen bereits im Speicher befindlichen Teil eines Programms ganz oder teilweise überschreibt und damit unbrauchbar macht. Dieses Verfahren wird benutzt, um umfangreiche Programme in einzelne, voneinander unabhängig ablaufende Unterprogramme aufzuteilen und den Speicherbedarf des Programms zu reduzieren. Dies kann besonders für den Amiga von Nutzen sein, da durch die Multitaskingfähigkeit der freie Speicherplatz für andere Programme zur Verfügung steht.

overlay file Eine Datei, die ein -> nachzuladendes Overlay-Modul enthält.

overscan Ein Bereich des Bildschirms, der in der Regel nicht für die Darstellung genutzt wird. Um sicher zu gehen, daß das zu erzeugende Bild vollständig und ohne Unschärfen am Bildschirm erscheint, ist bei jedem Fernseher/Monitor der Elektronenstrahl so justiert, daß dieser seitlich sowie oben und unten einen größeren Bereich als für die Darstellung notwendig »abfährt«. Beim NTSC-Amiga liegt der darstellbare Bereich normalerweise zwischen der 64. und 363. Spalte (LoRes) und zwischen der 44. und 243. Zeile. Der PAL-Amiga hat einen um 56 Zeilen erweiterten Darstellungsbereich.

Hardwaremäßig läßt sich die Lage des Bildes am Bildschirm innerhalb gewisser Grenzen ändern. Mit den Verschiebegadgets der -> Preferences kann der Anwender eine Einstellung vornehmen. Bei Verzicht auf die Sprites kann die horizontale Auflösung vergrößert werden (siehe horizontal blanking, vertical blanking, system timing frequencies).

packet Eine besondere Form von -> message, die von Amiga-DOS benutzt wird, um die Kommunikation zwischen Amiga-DOS-Tasks zu regeln. Insgesamt gibt es 21 verschiedene Typen (Version 1.1), die teilweise eng mit CLI-Befehlen verbunden sind. Eine ausführliche Beschreibung befindet sich im entsprechenden Kapitel des Amiga-DOS-Manual.

paper size (bei Preferences) -> spacing

PAR: Die parallele Schnittstelle. Voreingestellt ist diese Schnittstelle für die Übertragung von Daten an einen mit Centronics-Anschluß ausgerüsteten Drucker.

parallel.device Die Systemroutinen für die Verwaltung der parallelen Schnittstelle. Unter Version 1.1 kann es bei gleichzeitigem Betrieb der seriellen Schnittstelle zu Störungen an der Parallel-Schnittstelle kommen, weil das DRDY-Signal des parallelen Ports vom seriellen Port beeinflußt wird. Unter Version 1.2 soll dieser Mangel behoben worden sein.

pass through connector Ein Stecker, der einen belegten Anschluß am Computer nicht blockiert, sondern »durchschleift«.

path Suchpfad des -> filing systems. Die volle Namensbezeichnung einer Datei: Laufwerksnummer, erforderliche Verzeichnisnamen und Dateinamen.

Unter Workbench Version 1.2 läßt sich mit dem Befehl PATH der Suchpfad des CLI erweitern. Der voreingestellte Pfad für CLI-Befehle umfaßt das Hauptverzeichnis (beziehungsweise aktuelle Verzeichnis) und das Verzeichnis c. Der Befehl EXECUTE sucht Batch-Dateien zuerst im Hauptverzeichnis oder aktuellem Verzeichnis und dann im Verzeichnis s. Bei jedem anderen Dateizugriff wird ohne Angabe des vollständigen Pfades die Datei im Hauptverzeichnis erwartet.

PATH SHOW gibt den aktuellen Suchpfad auf dem Bildschirm aus. PATH allein stellt den Standardsuchpfad wieder her. Mit PATH ADD und dem gewünschten Verzeichnisnamen wird der Suchpfad entsprechend erweitert. Die Einstellung kann durch weitere PATH ADD Anweisungen ständig ergänzt werden.

pattern matching Der Vergleich von Zeichenketten anhand eines vorgegebenen Musters. Im CLI werden zwei solche Optionen angeboten. Einmal mit dem -> keyword p für einzelne Zeichen (?) oder eine beliebige Anzahl desselben Zeichens (#) und s für eine bestimmte Zeichenfolge (siehe special characters).

Paula (auch Portia) Der »peripheral/audio«- oder Port-Baustein unter den Custom-Chips. Dieser Baustein wandelt im Speicher abgelegte digitale Werte in elektrische Schwingungen um, die bei Wiedergabe über einen Lautsprecher entsprechende Klänge ergeben. Mit Paula läßt sich die gespeicherte Wellenform (Klangfarbe) beliebiger Instrumente abspielen. Um die Musikqualität zu erhöhen, ist ein störende Geräusche entfernender Tiefpaßfilter eingebaut. Der Chip hat vier Soundkanäle, von denen jeweils zwei zu einem Stereoausgang zusammengefaßt werden.

Eine weniger bekannte Aufgabe von Paula ist die Verwaltung der Hardwareschnittstellen (Parallel-Port, Seriell-Port, ...) des Amiga.

period Die Periode oder Dauer eines Signals (Tons). Diese Dauer steht im umgekehrten Verhältnis zur Frequenz, die erzeugt werden muß. Der Periodenwert eines Tons wird vom -> Copper in die Register der Audiokanäle geschrieben und dort als Zählerwert benutzt. Ausgehend von diesem Wert wird nach Null heruntergezählt. Die dabei benötigte Zeit beeinflußt die Tonhöhe. Je größer der Periodenwert, um so niedriger der erzeugte Ton. Eine verzerrungsfreie Tonerzeugung erfolgt mit Periodenwerten zwischen 124 (der niedrigstmögliche Wert) und 256. Werte über 256 sind zwar möglich, aber wegen der Beeinflussung durch den Tiefpaßfilter kommt es zu »aliasing distortion« (siehe system timing frequencies).

physical device Ein-/Ausgabeoperationen verrichtende Hardwarekomponenten des Amiga. Amiga-DOS kennt folgende physical devices: DF:, DH: (JH: bei der -> Janus-Schnittstelle), SER:, PAR:, PRT:, CON:, RAW:, NIL:, RAM: und »*«. Während die meisten devices nur Einzelgeräte sind, können bei den Laufwerken mehrere »device units« (Einheiten) angesprochen werden. Der Anwender spezifiziert die jeweilige Unit durch die Angabe einer Ziffer im Anschluß an den Namen des devices (DF0: (internes Laufwerk), DF1:, ...) (siehe logical device).

pinout (Pinbelegung) Die Verteilung der Anschlüsse für die Systemleitungen an einer Hardwareanschlußstelle des Computers. Während der A 500 und A 2000 Standardbelegungen bei der parallelen und seriellen Schnittstelle aufweisen, weicht der A1000 von der Norm geringfügig ab (siehe Anwender-Handbuch, Zorro).

pipelining Ein Verfahren zur Behebung zeitlicher Engpässe bei der Ausführung von Anweisungen (Beispiel: Die Ausführung einer Datenverarbeitungsoperation dauert so lange, daß zwischenzeitlich neue Eingabewerte anstehen). Da Datenverarbeitungsoperationen meistens in verschiedene Einzelschritte zerlegt werden können, wird nur ein Teil des ganzen Vorgangs von dem eigentlich zuständigen (Sub-)System erledigt und der Rest an ein anderes (Sub-)System für den Abschluß der Operation weitergegeben. Damit

können neue Datenwerte ohne größeren Zeitverlust verarbeitet werden. Das Verfahren wird unter anderem bei Agnus benutzt, führt aber dazu, daß das erste Datenwort bei -> DMA-Operationen verlorengeht. Ein ähnliches Verfahren auf höherer Ebene stellt das MetaComCo-TOOLKIT mit dem Befehl PIPES zur Verfügung.

pitch 1 Die Zeichendichte eines Zeichensatzes. Die Dichte der Zeichen bestimmt die Anzahl der Zeichen, die sich auf einem bestimmten Zeilenabschnitt — in der Regel 1 Zoll (2,54 cm) — unterbringen lassen. Mit -> Preferences sind drei Werte wählbar: 10 cpi (characters per inch: Zeichen pro Zoll) oder Pica, 12 cpi (Elite) und 15 cpi (Fine) mit jeweils 80, 96 und 120 cpl (characters per line: Zeichen pro Zeile). Diese Angaben gelten für ein Standardformular mit einer Breite von 8 Zoll.

pitch 2 Die Höhe (Frequenz) eines Tones. Gemessen wird diese in Hertz (Anzahl Schwingungen pro Sekunde, siehe period).

pixel (picture element: Bildpunkt) Die Ausgangsgröße eines Bildpunktes wird beim Amiga im niedrigauflösenden Modus bestimmt. Demnach ist ein Bildpunkt mit den Abmessungen $\frac{1}{320}$ der Breite und $\frac{1}{200}$ (NTSC) beziehungsweise $\frac{1}{256}$ (PAL) der Höhe des dargestellten Bildes definiert. Bei hochauflösendem Modus beträgt die Breite $\frac{1}{640}$ der Bildbreite. Das »aspect ratio« eines Pixel (Verhältnis Höhe zu Breite) ist also 8:5 beziehungsweise 4:5 für NTSC und 5:4 beziehungsweise 2,5:4 für PAL.

Ein Pixel auf dem Bildschirm entspricht einem Bit einer -> Bitplane. Ist das Bit gesetzt, wird das Pixel in einer Vordergrundfarbe abgebildet. Ein gelöschtes Bit bewirkt die Darstellung in der Hintergrundfarbe.

PlanePick/PlaneOnOff -> mask 1

playfield mode Die Zusammenfassung mehrerer Bitplanes für eine Darstellung am Bildschirm. Anders als beim -> display field ist ein Playfield nicht an die Abmessungen des darstellbaren Bildschirmbereichs gebunden. Es kann sowohl höher als auch breiter sein. Man unterscheidet Single-Playfields und Dual-Playfields. Während Single-Playfield der übliche Darstellungsmodus ist, lassen sich mit Dual-Playfield spezielle Effekte erzeugen. So kann ein Playfield zum Beispiel die Bitmuster für das Kontrollpanel eines Flugzeuges enthalten, während aus den Planes des zweiten Playfields die Bildinformation für die umgebende Landschaft — also die Sicht aus dem Cockpitfenster — geholt wird. Durch horizontales und/oder vertikales Scrollen nur der zweiten Plane ergibt sich der Eindruck einer Bewegung des Flugzeugs.

Im Dual-Playfield-Mode ordnet das System dem ersten Playfield die Bitplanes mit den ungeraden Nummern (1, 3 und 5) und dem zweiten die restlichen (2, 4, 6) zu. Da sich die Playfields auch die Farbregister teilen müssen, stehen für jedes nur noch 16 Farben zur Verfügung. Im hochauflösenden Modus sind für jedes Playfield nur zwei Bitplanes verwendbar. Hier halbiert sich also die Anzahl der verfügbaren Farben ein weiteres Mal. Das Besondere am Dual-Playfield-Mode ist die gleichzeitige Darstellung beider Playfields in einem Fenster (Window), wobei diese unabhängig voneinander verschoben werden können. Die Priorität der Playfields untereinander sowie gegenüber den Sprites läßt sich je nach Bedarf bestimmen. Verweist die Bitkombination eines Pixels des Playfields im Vordergrund auf das Hintergrundfarbregister (alle Bits 0), so erscheint der Punkt in der Farbe, die durch das zweite Playfield bestimmt wird. Damit läßt sich der oben erwähnte »Blick durchs Fenster« realisieren.

polling Die Abfrage von Systemeinheiten nach einem bestimmten Ereignis. So kann der Prozessor zum Beispiel durch Lesen bestimmter Register (Speicherstellen) der Ein-/Ausgabebausteine erkennen, ob ein Datenwert zur Eingabe bereit steht. Oft ist eine solche Abfrage in der Form einer Endlosschleife realisiert, die erst verlassen werden kann, wenn das gewünschte Ereignis eingetreten ist (auch »busy wait« oder »busy loop« genannt).

Dieses Verfahren ist beim Amiga unüblich, weil durch das laufende Abfragen wertvolle CPU-Zeit verlorengeht. Während der Prozessor nämlich für ein Programm auf bestimmte Ereignisse wartet, kann er genausogut andere Programme bearbeiten.

Der Amiga verwendet deshalb das sogenannte -> event trapping. Der Anwender definiert das Ereignis, auf das reagiert werden soll, und schickt das jeweilige Programm mit der Systemfunktion »wait()« in die Wartestellung. Diese Deaktivierung läßt sich in Amiga-Basic mit dem Befehl SLEEP auslösen. Dabei verliert das Programm den Zugang zu den -> resources. Es bleibt inaktiv, bis das gewünschte Ereignis eintritt. Das Programm wird »geweckt« und von -> exec wieder in das -> ready queue (die Schlange der arbeitenden Programme) eingeordnet. Durch solche Verfahren können andere Routinen während dieses »Nickerchens« fast unbehindert weiterarbeiten.

Port 1 Hardware-Port (auch Schnittstelle): Die Bezeichnung einer Anschlußstelle für Peripheriegeräte. Man unterscheidet parallel port, serial port, drive port (Laufwerk), mouse port und printer port.

port 2 Software-Port: -> message port.

potgo.library Systemroutinen für die Verwaltung von Eingaben über die Anschlüsse des game ports.

preemptive allocation Die Zuweisung einer -> resource an den -> task mit der zur Zeit höchsten Priorität. Das bedeutet die Übernahme einer resource, bis ein anderer task mit höherer Priorität aktiviert wird, oder die dem task gestattete Benutzungsfrist (-> quantum) abgelaufen ist (siehe timesharing). Meldet sich ein task mit der höchsten Priorität (127), übernimmt er die resource voll und ganz. Damit ist Multitaskingfähigkeit nicht mehr möglich.

Preferences (wörtlich »Vorlieben«; laut Anwenderhandbuch »der Voreinsteller«) Eine Sammlung von Systemroutinen, mit denen sich verschiedene Systemparameter einstellen lassen. Für die Änderung der Workbenchfarben stehen mehrere Schieberegler zur Verfügung, mit denen die Rot-, Grün- und Blauanteile der durch Anklicken gewählten Farbe bestimmt werden können. Mit weiteren Schaltern läßt sich Datum und Uhrzeit, Bildschirmauflösung (40/80 Zeichen, Interlace) oder das Zeitverhalten bei Mausklicks bestimmen. Mit einem Pixeleditor kann für spezielle Anwendungen ein neuer Mauszeiger entwickelt werden. Benutzer der seriellen Schnittstelle können mit Change Serial die Übertragungsgeschwindigkeit (Baud-Rate) und weitere Parameter für die serielle Datenübertragung einstellen. Change Printer ermöglicht die Einbindung einer Druckerparameterdatei in das System. Jede Parameterdatei enthält die Steuersequenzen (für Fettdruck, Breitschrift, Umlaute oder ähnliches) für einen bestimmten Drucker. Durch eine solche Anpassung können Programme unabhängig von der Arbeitsweise des Druckers entwickelt und eingesetzt werden. Mit der Papierlänge, dem rechten und linken Rand lassen sich weitere Einstellungen für die Druckausgabe vornehmen.

Für den Grafikdruck sind Angaben über shade, aspect, image und treshold wichtig. Für shade (Schattierung) haben Sie die Wahl aus: black and white (schwarz auf weiß), gray scale (Graustufen) und color (Farbe). Geben Sie »schwarz und weiß« an, so werden ab einer bestimmten Helligkeitsstufe der Punkte »weiße« statt schwarze Punkte ausgegeben. Diese Grenze läßt sich mit treshold einstellen. Eine andere Art der Farbumsetzung ist die Darstellung der Farben durch unterschiedliche Punktdichten (gray scale). Die Einstellung color ist nur für Farbdrucker. Mit aspect (Orientierung des Ausdrucks) kann der sonst horizontale Ausdruck um 90 Grad gedreht werden. Die Wahl des image bestimmt schließlich die Art des Ausdrucks: positiv oder negativ. Bei negativer Ausgabe erscheinen helle Flächen im Bild dunkel und dunkle Flächen entsprechend hell. Die-

se Einstellung läßt sich nur bei schwarzweißer Schattierung aktiviern.

Verlassen Sie Preferences mit CANCEL, ignoriert das System eventuelle Änderungen der Parameter. Drücken Sie auf SAVE oder USE, werden alle Änderungen sofort gültig (außer Bildschirmauflösung) und bei SAVE zusätzlich noch in der Datei »system configuration« gespeichert.

printer.device Die Schnittstellen-Routinen für die Druckausgabe. Diese Routinen übersetzen mit Hilfe einer im Verzeichnis »devs/printers« abgelegten Datei die Steuersequenzen des Amiga in druckerspezifische Steuersequenzen. Die jeweilige Parameterdatei kann mit dem Programm -> Preferences bestimmt werden. Die Liste der Amiga-Sequenzen befindet sich im -> RKM (Rom Kernel Manual) Libraries & Devices, Seite 420/421, Seite E41/E42 oder in den -> include files »printer.i« beziehungsweise »printer.h«.

PRT: Das DOS-Device PRT: (Drucker) des Amiga entspricht dem Gerät LPRT: bei MS-DOS und kann auch als solches von Amiga-Basic aus angesprochen werden. Der Unterschied zwischen PRT: und -> PAR: ist allerdings nicht nur bei Amiga-Basic wichtig: als (L)PRT: kann das Gerät druckerspezifische Steuersequenzen nicht verstehen, in Programmen muß man die Amiga-Sequenzen verwenden. Will man aber die Druckfunktionen unbedingt mit den druckerspezifischen Sequenzen ansteuern, muß der Drucker über PAR: oder SER: angesprochen werden. Außerdem wandelt PRT: die Zeilenendemarkierung des Amiga (Code 10: Linefeed) in die Standardsequenz Wagenrücklauf (Code 13) und Zeilenvorschub (Code 10) um. Eventuell muß der Drucker entsprechend eingestellt werden, sonst erhält man einen doppelzeiligen Abstand bei Texten und Grafiken.

priority queue Nach Priorität geordnete verkettete Liste des Systems (siehe ready queue, timesharing).

process Ein im Ablauf befindlicher Codeblock. Ein process ist mehr als ein -> task. Er hat einen eigenen Speicher, nutzt »eigene« -> resources und funktioniert auf einer höheren Systemebene. Tasks werden von -> exec, processes von -> Amiga-DOS verwaltet.

Ein process besteht unter Umständen aus mehreren tasks und kann als »parent process« auch neue processes erzeugen und starten (daughter processes). Ein Beispiel dafür ist das CLI. Jedes mit NEWCLI oder RUN gestartete CLI ist ein »Tochterprozeß« des ersten CLIs. Da aber tasks die Grundlage eines process sind, richtet sich die Priorität des process nach derjenigen des tasks.

program library Eine Sammlung von lauffähigen Routinen, die von einem bestimmten Programm als »run time library«, also während der Programmausführung, benötigt werden. Ein Beispiel hierfür sind die meist mit der Endung ».lib« versehenen Bibliotheken von Compilern oder Assemblern.

programming environment -> operating system

project (Arbeitsprojekt) Das unter Verwendung eines Programms angestrebte »Endprodukt«, welches nach Abschluß des Programmablaufs als Datei vorliegt (ein Dokument, Bild, Musikstück oder ähnliches).

protect tab (auch write-protect tab, read/write tab) Die Schreibschutzlasche einer Diskette. Während bei einer Diskette mit 3½ Zoll Durchmesser der Schreibschutz bei geöffnetem Loch aktiviert ist, muß bei einer 5¼-Zoll-Diskette der Ausschnitt zu diesem Zweck abgeklebt werden.

public domain-software Echte Public Domain-Software wird vom Programmautor der Allgemeinheit gebührenfrei, frei kopierbar und frei verwendbar (freeware) zur Verfügung gestellt. Es werden keine Copyright-Rechte geltend gemacht. Die meisten Autoren behalten sich allerdings vor, daß bei der Weitergabe oder Weiterentwicklung der Namensvermerk nicht gelöscht werden soll. Mittlerweile gibt es etwa 400 Disketten mit Public Domain-Software.

Eine andere Form der Public Domain ist die sogenannte »shareware«. Hier bittet der Autor um einen in der Regel eher symbolischen Betrag für die Nutzungsrechte an seinem Programm. Dafür stellt er gegebenenfalls dem »registrierten« Benutzer verbesserte Versionen und/oder Bedienungshilfen und Dokumentation zur Verfügung. Im Gegensatz zu »freeware« darf »share ware« eigentlich nur mit Genehmigung des Autors entgeltlich weitergegeben, oder in geänderten Versionen/Programmen von anderen benutzt werden.

Für den Amiga gibt es neben den -> Fish Disks mehrere Sammlungen mit solchen Programmen: FAUG, Amicus, AMuse, Phase4, HotMix, BCS, Software Digest (alle USA), Amiga Users' Group und ICPUG (UK). Weitere Software kann von amerikanischen bulletin boards per Modem abgerufen werden. Es empfiehlt sich allerdings, diese Angebote vorher zu überprüfen, weil viele mit dem Inhalt mancher Fish Disk übereinstimmen beziehungsweise Routinen von Fish Disks in einer neuen, meist themenorientierten, Zusammenstellung anbieten. Da der Vertrieb solcher Software nicht als Gewinngeschäft gedacht ist, sollte der Preis nur die Kosten für Diskette und Zeitaufwand des Kopierens abdecken. Preise von mehr als 8 Mark grenzen schon an Gewinnsucht. Mehr als 10 Mark ist Wucher.

pulse code modulation Ein Sammelbegriff für verschiedene Formen der Signalmodulation, die insbesonders bei der En- und Decodierung von Daten bei der Übertragung an externe Speichermedien verwendet werden. Die Grundlage des Verfahrens ist das Umkehren der magnetischen Polarität während eines »clock pulse« (siehe system timing frequencies), um den binären Wert des aktuellen Signalpulses zu codieren. Auf dem Amiga spielt das Verfahren beim -> trackdisk.device eine besondere Rolle, da es so umprogrammiert oder durch Zusatzroutinen erweitert werden kann, daß verschiedene Aufzeichnungsformate (MFM (Commodore), GCR (Apple), IBM) gelesen und geschrieben werden können.

quantization Ursprüngliche Bezeichnung für das Umwandeln eines analogen Signals in digitale Form. Heute unterscheidet man »Digitalisieren« (Audioquantisierung) und »Pixelisieren« (Bildquantisierung). Übriggeblieben ist nur die Bezeichnung »quantization noise« für Störgeräusche im Signal, die sich besonders bei niedriger Lautstärke des Signals und niedriger Samplingfrequenz bemerkbar machen. Diese Störungen entstehen meist, weil die einzelnen »Stufen« des digitalisierten Signals eine zu grobe Annäherung an das Originalsignal darstellen. Die Stellen, an denen gemessen wird (»hold phases«), sind in der Regel zu weit auseinander, um den richtigen Verlauf der Signalschwingung voll zu erfassen (siehe Nyquist sampling, digitizer).

quantum (Menge) Ein zugewiesener Anteil. In diesem Fall ein Anteil an der Benutzung der -> system resources. Jeder -> task erhält beim Amiga ein Quantum von 4/60 Sekunden. Das ist die Zeit, die benötigt wird, um eine normale (NTSC) Bildschirmdarstellung viermal (im Interlace-Modus zweimal) zu erzeugen (siehe timesharing, system timing frequencies).

queue (Warteschlange) Eine besondere Form der -> linked list. Beim Amiga spielen drei queues eine Rolle: ready queue, waiting queue und message queue.

Ready queue: Eine nach Priorität sortierte Liste der Tasks, die auf ihr -> quantum warten. Die Tasks warten nicht auf Eingaben, sondern auf die Möglichkeit, vorliegende Daten zu verarbeiten.

Waiting queue: Diese Liste enthält diejenigen tasks, die auf eine Eingabe (ein Ereignis) warten.

Mit beiden Warteschlangen steuert das Multitasking-System den Zugriff der einzelnen tasks auf die -> resources des Amiga.

Message queue: Diese Warteschlange entsteht, wenn an einem -> message port die eingehenden messages nicht schnell genug verarbeitet werden (siehe system log jam).

Der Pseudo-Guru

Die letzte Aufgabe war schon etwas schwieriger als die vorangegangenen, aber mit etwas Geduld und Fingerspitzengefühl haben Sie die Fehler sicher gefunden. Heute stellen wir Ihnen den ersten Pseudo-Guru in der Sprache C vor. Damit kommen auch die Benutzer dieser »gefährlichen« Sprache nicht zu kurz.

Im letzten Programm (Ausgabe 1/88) waren mehrere kleine Wanzen (Bugs) versteckt, die nun enttarnt werden. Aus diesen absichtlich mit Fehlern versehenen Programmen lernen Sie im Laufe der Zeit, Fehler zu vermeiden. Falls sich dann doch noch einer einschleicht, werden Sie ihn sicher schneller finden, da Sie im »Debugging« (Fehlersuche) schon ein richtiger Profi sind. Denn Übung macht den Meister.

Bei dem letzten Programm muß als erstes die zweite Zeile gelöscht werden, da die Funktion »inputstring« später aufgerufen wird. Bei der Definition der drei Unterprogramme müssen die formalen Parameter (a$ und x$) natürlich in Klammern stehen, sonst erscheint der Fehler »Argument count mismatch« beim Aufruf der Funktionen. Das kommt daher, daß der Basic-Interpreter bei den fehlenden Klammern nicht erkennt, wie viele Argumente benötigt werden. Einzelheiten zu den Unterprogrammen können Sie in Ihrem Basic-Handbuch im Kapitel 6 - 1 nachschlagen.

In der Funktion »getstring« wird versucht, mit dem Befehl INPUT den Text einzulesen. Dies funktioniert, solange keine Leerstelle oder Komma im Text enthalten ist. Damit auch diese zwei Zeichen enthalten sein dürfen, muß die Zeile zum Einlesen des Textes folgendermaßen lauten:

```
LINE INPUT #1,z$
```

Dieser Befehl liest alle Zeichen aus einer sequentiellen Datei bis ein Zeilenendekennzeichen (Carriage Return) erreicht wird.

Um das Maß vollzumachen, hat der Guru auch noch eine Zeile gelöscht. Damit die Funktion »inputstring« richtig arbeitet, muß vor der zweiten WHILE-Schleife noch die Zeile

```
c$=""
```

eingefügt werden. Dies ist sehr wichtig und wird sehr oft vergessen. Vor WHILE-Schleifen muß zuerst die Variable richtig gesetzt werden, die in der Abbruchbedingung benutzt wird. In unserem Fall wird die innere WHILE-Schleife nämlich nur einmal durchlaufen, weil danach die Variable c$ einen Wert besitzt und die Abbruchbedingung beim zweiten Durchlauf schon erfüllt ist.

Da es soviele Fehler waren, hier noch einmal die lauffähige Version des Programms:

```
INPUT"Dateiname : ";datei$
putstring datei$
getstring datei$
END

SUB putstring (a$) STATIC
 OPEN a$ FOR OUTPUT AS #1
 inputstring x$
 PRINT#1,x$
 CLOSE 1
END SUB

SUB getstring (a$) STATIC
 OPEN a$ FOR INPUT AS #1
 LINE INPUT#1,z$
 PRINT z$
 CLOSE 1
END SUB

SUB inputstring (x$) STATIC
 PRINT "Bitte Text eingeben::"
 FOR i=0 TO 50
  WHILE (c$<>CHR$(13))
   c$=""
   WHILE(c$="")
    c$=INKEY$
    PRINT c$;
   WEND
   x$=x$+c$
  WEND
 NEXT i
END SUB
```

Nach soviel Fehlern in Basic-Programmen kommt diesmal ein kurzes C-Listing, das nur zwei »Bugs« enthält. Auch hier ist es wieder ratsam, sich Unterlagen wie etwa Bücher über die Programmiersprache C bereitzulegen. Aber auch der C-Kurs in unserem Magazin hilft Ihnen sicher weiter.

```
void main()
{
  int a,b,summe;

  printf("\fErste Zahl  : ");
  scanf("%d",a);
  printf("\nZweite Zahl : ");
  scanf("%d",b);
  printf("\nSumme = %d",a+b);
}
```

Übersetzen Sie das Programm mit den folgenden Befehlen für den Aztec-C-Compiler:

```
cc summe +l
ln summe.o -lc32
```

Besitzen Sie den Lattice-C-Compiler, verwenden Sie die Aufrufe:

```
lc summe
blink lib:c.o,summe.o lib lib:lc.lib,lib:amiga.lib
```

Lesen Sie das Listing sehr genau durch, denn nur »wer suchet, der findet«. Dies gilt insbesondere bei den C-Compilern, die eigentlich sehr wenig Fehler anzeigen. Dies liegt daran, daß in C »alles« erlaubt ist. Deswegen ist in C auch eine strenge Disziplin beim Programmieren nötig, um von vornherein Fehler soweit als irgend möglich auszuschließen. Meist werden nämlich die verhängnisvollen Fehler nicht in Formeln oder ähnlichem gemacht, sondern bei den »kinderleichten« Anweisungen. Man weiß ja sowieso genau wie es geht und dabei vergißt man dann leicht ein Zeichen oder macht eines zuviel. Ein gutes Hilfsmittel dagegen ist eine eigene Bibliothek mit kleinen, universellen Routinen. Diese Funktionen können gut getestet werden und sind dann (ohne Fehler) leicht immer wieder verwendbar. Ein gutes Beispiel dafür ist der »File Requester« aus Ausgabe 1/88 auf Seite 94. Setzen Sie ihn ruhig immer wieder ein, Ihre Programme werden dadurch sicherer. (rb)





Das MIDI Praxisbuch

»Das MIDI Praxisbuch« — geschrieben zu einem Zeitpunkt, als der Amiga noch regulär 6000 Mark kostete — geht relativ wenig auf das Thema Amiga und Midisteuerung ein. Der Amiga war damals noch zu neu, die Informationen über diesen »Wundercomputer« zu verworren und Midi-Software kaum verfügbar. Wenn in diesem Buch von Mikrocomputern die Rede ist, dann sind es der C 64, die Atari ST-Serie und der Macintosh.

Der Leser wird systematisch in die Geheimnisse des Midi-Standards eingeführt. Es sind keinerlei Grundkenntnisse für das Studium dieses Buches erforderlich. Eindeutig im Vordergrund steht der Musiker. Das Midi-Praxisbuch enthält nur wenige Programmierhinweise oder gar Listings. Vielmehr erfährt der Leser von Grund auf alles über Midi-Normen, Anschlußbelegungen und eine ganze Menge praktischer Tips und Tricks im Umgang mit Synthesizern und anderen Peripheriegeräten.

Der erste Teil des mit 400 Seiten sehr umfangreichen Buches ist ein Grundlagenkapitel über Midi selbst. In einem weiteren Teil bietet der Autor eine Marktübersicht. In dieser Übersicht wird jedes Gerät erwähnt, das in irgendeiner Weise mit Midi umgehen kann. Diese schon fast einem Katalog ähnliche Aufzählung aller Midi-Geräte birgt natürlich auch die Gefahr in sich, daß das Buch schnell unaktuell wird, da der Musikmarkt sich genauso rasend ändert wie der Computermarkt. Der Leser erhält jedoch einen guten Überblick über die verschiedenen »Musikinstrumente«, die mit Midi kontrolliert werden können. Der dritte Teil befaßt sich mit Midi-Spezifikationen und den zugehörigen Reglementierungen. Im vierten Teil geht es dann ans Eingemachte: Der Autor stellt Midi-Studio, Sequenzer, Synchronisationsprobleme und spezielle Software zur Diskussion.

Die Zielgruppe des Praxisbuches sind eindeutig die Musiker. Es sind weder Midi- noch Computervorkenntnisse nötig. Die Musikprogrammierer unter den Amiga-Fans können allenfalls die Grundlageninformationen verwerten. Wer sich ein Midistudio aufbauen möchte oder live auf der Bühne Midi einsetzen will, ob mit oder ohne Amiga, wird in diesem Buch wichtige Informationen finden.

(Joschy Polierer/pa)

Das MIDI-Praxisbuch, Richard Aicher, Signum Medien Verlag GmbH, 400 Seiten, ISBN 3-924767-12-2, Preis: 48 Mark

Amiga Intern

Mit dem vorliegenden Buch liegt ein weiteres Werk aus der Intern-Reihe von Data Becker vor. Der Schwerpunkt von »Amiga Intern« liegt mit 300 Seiten eindeutig bei der Hardware des Amiga und ihrer Programmierung. Ein Drittel davon ist ausschließlich der Beschreibung der Hardware und ihrer Verwendung im Amiga gewidmet. Intensiver Gebrauch von Blockschaltbildern, Tabellen und Pinbelegungsdiagrammen erleichtern das Verständnis. Obwohl den Autoren eine verständliche Darstellung der komplizierten Materie gelungen ist, werden Einsteiger in die Interna ihres Computers wohl an manchen Stellen mit Verständnisproblemen zu kämpfen haben.

Der sehr ausführliche Abschnitt über die direkte Programmierung der Hardware in Assembler befaßt sich natürlich mit den drei Customchips im Amiga. Der Assembler-Programmierer findet hier eine Fülle von Informationen, die er bisher vergeblich gesucht hat.

Assembler und Multitasking sind gute Voraussetzungen, einen blinkenden, roten Streifen am oberen Rand des Bildschirms zu erzeugen. Damit die Lust nicht zum Frust wird, beschreiben die Autoren in einem weiteren Teil den Umgang mit Multitasking. Zusammen mit den anderen Routinen von »Exec« also wichtige Informationen für die Programmierer mit »System«. Ein Supertip ist die Beschreibung einer Technik, mit deren Hilfe Programme auch nach einem Reset aktiv bleiben können.

Die letzten beiden Teile behandeln das Amiga-DOS und den Zugriff auf die Devices des Amiga. Die Grafik-Fans finden dort unter anderem mit der Beschreibung des IFF-Standards Informationen auf dem Weg zum eigenen IFF-Lader.

Insgesamt ist »Amiga Intern« trotz des nicht gerade niedrigen Preises für Assemblerprogrammierer unentbehrlich. Für diejenigen, die dem Hantieren mit Hardware-Registern lieber die Systemroutinen vorziehen und statt in Assembler in C oder gar Basic arbeiten, sind die Informationen dieses Buches eher grundlegend und weniger für die tägliche Praxis brauchbar.

(Andreas Lietz/pa)

Amiga Intern, Dittrich/Gelfand/Schemmel, Data Becker, ISBN 3-89011-104-1, Preis: 69 Mark



Amiga Assembler-Buch

Endlich ein Autor, der es versteht, mit einfachen Worten dem Anfänger wie dem Fortgeschrittenen ein sicherlich sehr schwieriges Thema verständlich und zugänglich zu machen. Das »Amiga Assembler-Buch« ist eine meisterhaft gelungene, einprägsame Einführung in das Reich der Bits und Bytes des Amiga.

Das Buch beginnt mit einer grundlegenden Einführung. Auf das sonst oft so langwierige Kapitel: »Wie schalte ich meinen Amiga ein?« wurde sinnvollerweise verzichtet. Der Autor erklärt alles Notwendige über Assembler, binäre Arithmetik und den Aufbau eines Computers. Der Einleitung folgt ein ausführlicher Test der drei wichtigsten Assembler-Übersetzer, die für den Amiga erhältlich sind. Der Autor begründet ausführlich die Wahl seines bevorzugten Assemblers. Schon bald wird der Leser merken, wie der Autor in großen (Wissens-)Schritten ein Kapitel nach dem anderen durchschreitet. Dieses Buch ist dennoch gut zum Selbststudium geeignet, denn der didaktische Aufbau ermöglicht es dem Leser schnell, seinen eigenen Wissensstand einzustufen und die Lücken zu erkennen. Stellenweise sind manche Erläuterungen allerdings so knapp verfaßt, daß noch weitere Literatur zu Rate gezogen werden muß. Dies ist sicherlich vom Autor beabsichtigt, da er grundlegende Kenntnisse einer Programiersprache wie Basic oder C voraussetzt. Vor der Anwendung dieses Buches sollte sich der Leser über die Bedienung des Amiga im klaren sein und Grundkenntnisse über CLI und Betriebssystem besitzen.

Der Autor ist nicht einer jener Assemblerfreaks, die erst einmal die Lösung eines Problems in Assembler propagieren. Vielmehr setzt er sich kritisch mit der maschinennahen Programmierung auseinander und zeigt deutlich Probleme auf, die in einer Hochsprache besser gelöst werden können. Ein großes Kapitel befaßt sich daher mit der Einbindung von Assemblerroutinen in Amiga-Basic-Programme.

Derjenige Leser, der dieses Buch intensiv durchgearbeitet hat, hat viel über den Amiga erfahren und solide Assemblerkenntnisse erworben. Der Autor wird den Leser sicherlich länger als ein Wochenende an dieser Lektüre festhalten. Das »Amiga Assembler-Buch« wird mit einer Diskette ausgeliefert, auf der alle beschriebenen Programmbeispiele enthalten sind. Ein Buch, das Mut macht, die wohl schwierigste Programmiersprache zu erlernen.

(Joschy Polierer/pa)

Amiga Assembler-Buch, Peter Wollschlaeger, Markt & Technik Verlag, 387 Seiten, ISBN 3-89090-491-2, Preis: 69 Mark

Faszination Programmieren

Einige der bekanntesten Programmierer unserer Zeit werden in diesem Buch vorgestellt. Die Palette reicht von Bill Gates, der Microsoft Basic programmierte, über Jonathan Sachs, den Schöpfer von LOTUS 1-2-3, bis zu Toru Iwatani, dem mit Pac Man ein Welterfolg gelang. Sie erfahren viel über die Persönlichkeit und den Programmierstil der Profis unserer Zeit.

(ub)

Faszination Programmieren, Susan Lammers, Markt & Technik Verlag, 430 Seiten, ISBN 3-89090-418-1, Preis: 49 Mark

Grafik auf dem Amiga

Spannende Computerspiele werden erst durch schnelle Bewegung kleiner grafischer Objekte reizvoll. Das Zauberwort für die Animation solcher reizvollen Figuren heißt: Sprites.

In der letzten Folge haben wir uns ausführlich mit Bitplanes, Playfields und der Farbwahl für einen Bildschirmpunkt (Pixel) befaßt. Einige der dort beschriebenen Begriffe werden uns in den folgenden Ausführungen wieder begegnen. Wenn Sie sich über die Bedeutung des einen oder anderen Begriffes nicht mehr ganz sicher sind, lesen Sie sich die Erklärungen in den ersten beiden Teilen des Grafik-Kurses noch einmal durch.

Dieses Mal beschäftigen wir uns mit den Sprites. In größeren Wörterbüchern findet man als Übersetzung für diesen Begriff die Wörter Geist oder Kobold. Man kann sich über die Qualität dieser Namensgebung sicherlich streiten, aber eines steht fest: Die kleinen Kobolde können einen Spieler schon ins Schwitzen bringen.

Sprites sind kleine, schnell bewegbare Objekte. Damit sie überhaupt schnell bewegt werden können, sind Sprites nicht Teil eines Playfields. Das bedeutet, daß die für die Darstellung des Bildes notwendigen Informationen sich nicht als Bitmuster in irgend einer Bitplane befinden. Der Amiga verwaltet diese Objekte in der Tat unabhängig von der Playfield-Hardware. Ein spezieller Teil der Hardware ist nur für diese Kobolde zuständig.

Das wohl bekannteste Sprite benutzen Sie ständig bei der Kontrolle Ihres Computers. Es handelt sich um den Mauszeiger. Vielleicht haben Sie sich schon einmal darüber gewundert, wie dieser Pfeil in der Regel völlig unbeeinflußt vom gerade laufenden Programm über den Bildschirm dirigiert werden kann. Eine eigenständige Kontrollhardware — die sogenannte Sprite-Hardware — vereinfacht die Handhabung solcher Objekte. Für die Steuerung der Sprites brauchen Sie allerdings keine wesentlich neuen Verfahren erlernen. Die Organisation dieser Hardware und damit der Aufbau von Sprites ist in den Grundzügen der Playfield-Hardware sehr ähnlich.

Der Amiga kann bis zu acht Sprites voneinander unabhängig verwalten. Ein Sprite besteht genau wie ein Playfield aus einer Folge kleiner Pixel oder auch Bildpunkte, die zusammen ein Bild ergeben. Da die Verwaltung von selbst kleinen Playfields eine ganze Menge Aufwand an Hard- und Software benötigt, leuchtet es ein, daß Sprites gegenüber einem Playfield gewissen Einschränkungen unterliegen. Die wesentlichen Beschränkungen betreffen die Größe und die Farbe.

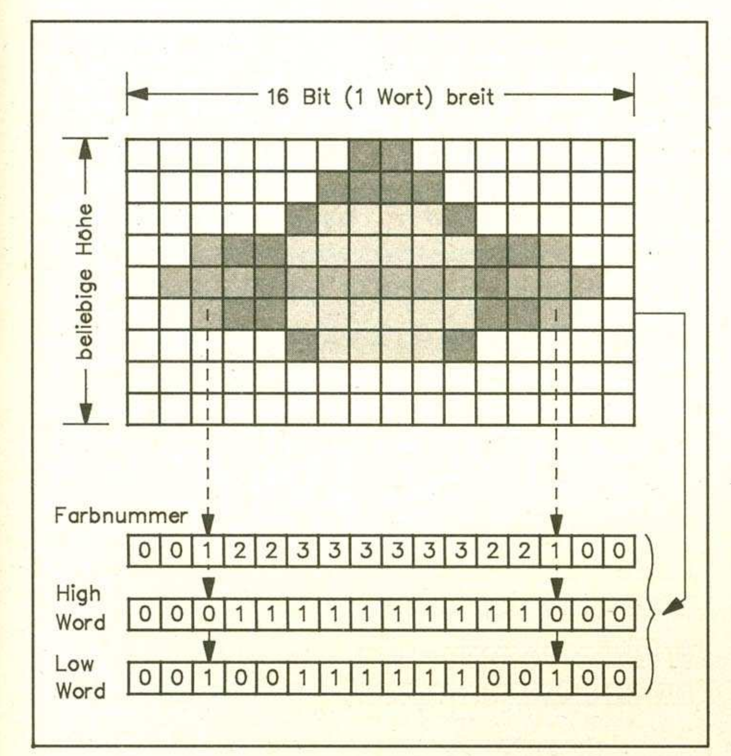


Bild 1. Der Entwurf eines Sprites. Für die verschiedenen Farben werden zugeordnete Werte eingetragen.

Die Bitplanes der Sprites

Für jedes Sprite steht ein DMA-Kanal zur Verfügung. Den Begriff DMA kennen Sie bereits als Bezeichnung für den direkten Speicherzugriff verschiedener Hardware-Bausteine. Ein Kanal ist nichts anderes als ein periodisch wiederkehrender Zeitabschnitt, währenddessen sich ein Teil der Hardware mit einer bestimmten Aufgabe — hier der Darstellung eines Sprites — beschäftigt. Nur im Laufe dieses Zeitabschnittes findet der jeweilige Hardwareteil einen »Kanal« zum Speicher, um sich notwendige Informationen zu besorgen oder anderen Schaltungen solche zukommen zu lassen. Auf diese Weise wird die Zeit, in dem unabhängig vom Prozessor auf den Speicher zugegriffen wird, auf verschiedene Komponenten (Audio-, Bitplane-, Sprite-DMA und andere) verteilt.

Die zur Verfügung stehende Zeit für ein Sprite reicht gerade aus, um 16 nebeneinanderliegende Pixel eines Sprites abzubilden. Eine Periode später sorgt der jeweilige DMA-Kanal für die Abbildung von weiteren 16 Punkten. Sie erkennen hier, daß die Länge einer solchen Periode genau der Zeitdauer entspricht, die der Elektronenstrahl für die Darstellung einer Bildschirmzeile benötigt. Sonst wären ja die 16-Pixel-Muster nicht genau untereinander. Konsequent weiter gedacht müssen wir zu dem Schluß kommen, daß ein Sprite beliebig hoch sein kann, da ja in jeder (Zeilen-)Periode Zeit für die Abbildung einer Spritezeile vorhanden ist. Das ist auch tatsächlich so. Ein Sprite kann also beliebig hoch, aber nur 16 Pixel breit sein.

Die Anzahl der möglichen Sprite-Farben ist wiederum aus zeitlichen Gründen auf drei verschiedene Farben beschränkt. Hinzu kommt noch als vierte »Farbe« Transparent. Belegen Sie ein Pixel mit dieser »Farbe«, so wird an dieser Stelle die Farbe angezeigt, die sich unter diesem (Sprite-)Pixel (ein Playfield oder andere Sprites) befindet.

Von der Beschreibung der Farbbildung bei Bitplanes wissen wir, daß sich vier verschiedene Farben nur mit Informationen aus zwei Bitplanes erzeugen lassen. Man kann sich daher ein Sprite als ein kleines, aus zwei Bitplanes bestehendes Playfield mit beliebiger Höhe und 16 Pixel Breite vorstellen. Von diesen Mini-Playfields kann der Amiga ohne weiteres nur acht Stück steuern, da nur acht DMA-Kanäle dafür zur Verfügung stehen. Mit ein paar Tricks können wir diese Beschränkung allerdings umgehen. Doch mehr Informationen dazu später.

Woher weiß der Amiga nun, wie ein Sprite aussehen beziehungsweise wo es sich auf dem Bildschirm befinden soll? Ganz einfach: Indem er sich an einer bestimmten Stelle des Speichers die notwendigen In-

Teil 3

KURSÜBERSICHT

In diesem Kurs stellen wir Ihnen die Arbeitsweise der für die Bilddarstellung wichtigen Chips des Amiga vor. In den einzelnen Folgen des Kurses finden Sie folgende Themen behandelt:

TEIL 1: (Einführung): Die Spezialchips im Amiga; Bilddarstellung durch Punkte; Arbeitsweise von Monitor und Fernseher; Farbbildung durch Bitplanes

TEIL 2: (Playfields): Die Playfield-Hardware; der Datenzugriff durch Denise; Dual-Playfields; Scrolling; Hold & Modify; Interlace

TEIL 3: (Sprites): Datenstruktur, Positionierung und Bewegung von Sprites; virtuelle Sprites; Sprite-Attach

TEIL 4: (Blitter/Copper): Arbeitsmodi des Blitter; Programmierung des Copper; die Kontrollhardware

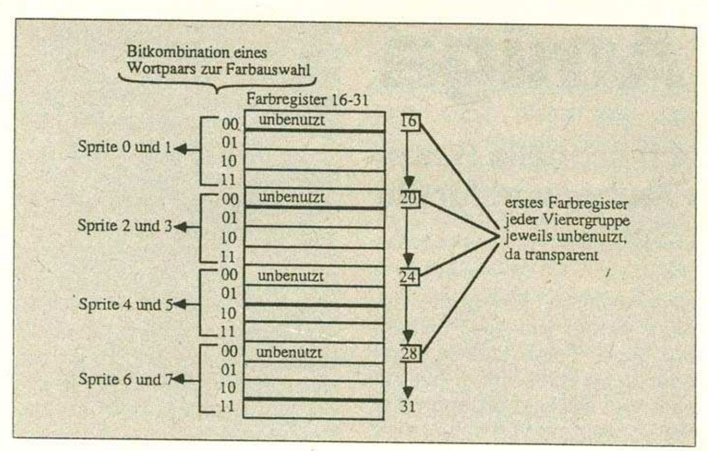


Bild 2. Die Zuordnung der Spritegruppen zu Farbregistern

formationen dazu holt. Bevor also ein Sprite auf dem Bildschirm erscheinen kann, müssen wir es erst mal definieren. In Bild 1 sehen Sie das von uns gewählte Beispiel. Es enthält Pixel in vier verschiedenen Farben (einschließlich transparent). Um die Farbinformationen nun für den Amiga aufzubereiten, können sie zunächst die Farbnummern jedes Pixel (0,1,2,3) in ein Gitter derselben Größe eintragen. Dann ersetzen Sie diese Nummern in die binären Kombinationen (0=00, 1=01, 2=10, 3=11). Auf diese Weise erhalten Sie für jede Zeile eines Sprites 32 Bit (= 2 x 16 Bit = 2 Worte = 4 Byte) Informationen. Man bezeichnet das Wort mit den Bits 0 bis 15 auch als niederwertiges, das mit den Bits 8 bis 15 als höherwertiges Wort.

Damit haben wir nun das Aussehen eines Sprites festgelegt. Die Farbwahl desselben wird also durch eine Folge von zwei Datenworten bestimmt, die durch Hintereinanderlegen zu einer Art Mini-Bitplane verknüpft werden. Hier finden Sie auch einen zweiten Grund für die Begrenzung der Breite auf exakt 16 Pixel: Ein Datenwort enthält 16 Bit. Zwei dieser Datenworte zusammen ergeben die Farbwerte der entsprechenden Spritezeile. Jeder Sprite-Punkt wird von einem 2-Bit-Wert dargestellt, der vier verschiedene Zustände einnehmen kann. Die Kombination 00 führt zu einem transparenten Pixel.

Leider besitzen die Sprites keine eigenen Farbregister. Es müssen vielmehr die Register der Playfield-Hardware mitbenutzt werden. Die Sprites nutzen allerdings nur die Farbregister 16 bis 31. Die Sprite- und Playfieldfarben kommen sich also erst ins Gehege, wenn Sie Playfields mit mehr als 16 Farben erzeugen.

Zwei aufeinanderfolgende Sprites teilen sich dieselben vier Farbregister. Das erste Register dieser Gruppe bleibt ungenutzt, da die auf dieses Register verweisende Bitkombination 00 ja immer ein transparentes Pixel erzeugt (Bild 2).

Nachdem das Aussehen unseres Sprites jetzt in Form und Farbe definiert ist, müssen wir eine Datenstruktur anlegen, die unter anderem diese Informationen enthält (Bild 3). Eine solche Datenstruktur besteht aus einer Reihe von Datenworten. Alle Daten, die zu einem Sprite gehören, müssen ohne Unterbrechung im Speicher abgelegt sein.

Die erste Information der Datenstruktur ist die Startposition des jeweiligen Sprites. Dafür sind zwei Datenworte vorgesehen. Man bezeichnet diese mit Sprite-Control-Word 1 und 2 beziehungsweise SPRxPOS und SPRxCTL. SPRxPOS enthält die Werte für die vertikale und horizontale Startposition des Sprites x. Die Bits 8 bis 15 bilden die 8 Bit der vertikalen Startadresse (VSTART). Es handelt sich hierbei jedoch nur um die niederwertigen Bits von VSTART. Da mit 8 Bit nur 256 Positionen abgedeckt werden können, unser Bildschirm aber höher ist, benötigen wir für VSTART noch ein höherwertiges Bit. Dieses befindet sich im Sprite-Control-Word 2 (SPRxCTL) und belegt dort Bit 2. Die Bits 0 bis 7 im Sprite-Control-Word 1 enthalten die höherwertigen 8 Bit für die horizontale Startadresse (HSTART). Das niederwertige Bit von HSTART befindet sich im Bit 0 des SPRxCTL-Wortes. Die restliche Belegung von SPRxCTL: Die Bits 3 bis 6 werden nicht genutzt und sollten auf Null gesetzt werden. Bit 7 ist ein spezielles Kontrollbit, dessen Funktion wir später noch behandeln. Die Bits 8 bis 15 geben zusammen mit dem höherwertigen Bit (SPRxCTL Bit 1) das vertikale Ende des Sprites an. Diese 9-Bit-Gruppe trägt die Bezeichnung VSTOP. Der Wert dieses Wortes wird jedoch nicht direkt an die Videologik übergeben. Auf den Wert von VSTOP addiert die Hardware den Wert VSTART+1. Für den Programmierer hat diese interne Addition den Vorteil, daß er sich nicht um die reale Endadresse des Sprites kümmern muß, sondern nur die Höhe des Sprites in VSTOP ablegen muß.

Die nächsten Worte der Datenstruktur enthalten die Informationen für die Bildpunkte der Sprites. Jeweils zwei Datenworte ergeben eine Zeile des Sprites. Das Ende der Bildinformationen erkennt die Videologik daran, daß das letzte Wortpaar nur Nullen enthält.

Grundsätzlich kann ein Sprite niemals über das mit DIWSTRT und DIWSTOP definierte Bildschirmfenster hinausragen. Es wird dann — unabhängig davon, ob das Bildschirmfenster kleiner ist als das vom Monitor darstellbare Bild — an dessen Grenze abgeschnitten.

Der erste Entwurf

Damit die Hardware nun ein Sprite automatisch auf dem Bildschirm anzeigt, muß die Sprite-DMA aktiviert werden. Dies geschieht durch Setzen von Bit 5 im Register DMACON. Vorher muß allerdings die Adresse der Datenstruktur in die beiden Register SPRxPTL und SPRxPTH geschrieben werden. SPRxPTL enthält den niederwertigen Teil (Bit 0 bis 15) und SPRxPTH den höherwertigen Teil (Bit 16 bis 18) der Datenstrukturadresse von Sprite x. Da die Custom-Chips nur auf die unteren 512 KByte des Amiga zugreifen können, ist diese Adresse auch nur 19 Bit breit.

Nach dem Einschalten der Sprite-DMA werden die Sprites einen Bildschirmaufbau lang angezeigt. Um Sprites ständig sichtbar zu machen, müssen während des vertikalen Blankings die Spritezeiger wieder auf den Startwert der Datenstruktur zurückgesetzt werden, da deren Inhalt sich im Laufe der Anzeige ändert. Bewegen können wir ein Sprite dann ganz einfach durch die Änderung der VSTART- beziehungsweise HSTART-Position innerhalb der Spritestruktur.

Da Sprites unabhängig von der darunter befindlichen Playfield-Grafik sind, werden alle Informationen, die ein Sprite betreffen, auf den LoRes-Non-Interlaced-Modus bezogen. Dies trifft also auch auf eine Darstellung vor einem Interlace- oder HiRes-Bild zu.

Die Startposition des Sprites wird von der oberen linken Ecke des Bildschirms aus auf die obere linke Ecke des Sprites bezogen. Die obere linke Ecke des Bildschirms hat die Koordinaten X=0 und Y=0. Wir wissen aber, daß nicht der gesamte Bereich des Bildschirms für die Darstellung genutzt werden kann. Durch das »Overscan« ist die frühestmögliche Startposition eines Sprites erst auf den Koordinaten X=64 und Y=44. Die maximal nutzbare Bildschirmbreite beträgt 384 Pixel. Der gesamte sichtbare Bereich des Bildschirms ist dann 320 Bildpunkte breit. Da ein Sprite 16 Pixel in Anspruch nimmt, und von der linken oberen Ecke ausgegan-

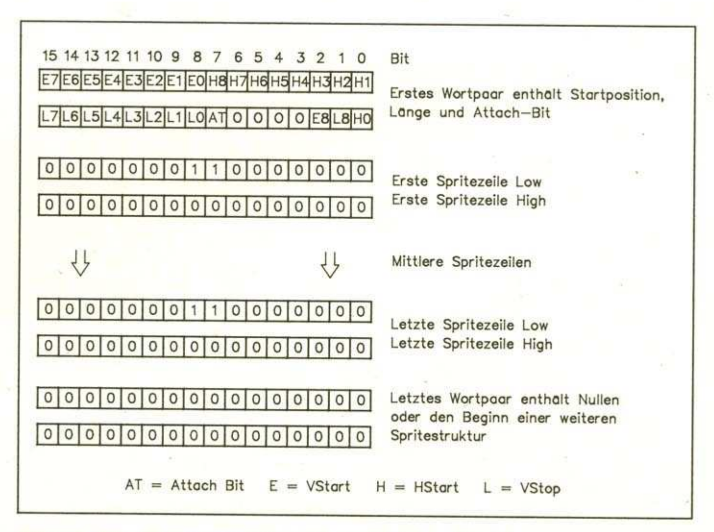


Bild 3. Die Datenstruktur für beliebige Sprites

gen wird, müssen diese 16 Pixel nochmals vom rechten Rand abgerechnet werden. Auf der so errechneten Position wird der rechte Rand des Sprites gerade noch nicht abgeschnitten. Diese Werte sind natürlich auch von der Justage des Bildausschnittes am Monitors abhängig. Wollen Sie ein Sprite ohne Beschneidungen auf einem bestimmten horizontalen Punkt des Bildschirms erscheinen lassen, addieren Sie den Wert 64 zu der gewünschten Startposition. Für die vertikale Position ist eine ähnliche Rechnung notwendig. Hier müssen jedoch 44 Zeilen zur gewünschten Startposition hinzugerechnet werden.

Ein Sprite ist übrigens immer 16 Pixel breit. Wenn Sie Sprites erzeugen wollen, die schmaler als diese 16 Pixel sein sollen, so müssen Sie die nicht benötigten Pixel in der Farbe »Transparent« ausfüllen. An diesen Stellen wird dann bei der Darstellung die Hintergrundfarbe durchscheinen. Mit den verbleibenden drei Farben können Sie dem Sprite jede beliebige Form geben, die in den vordefinierten Rahmen paßt (Bild 1).

Wie schon zu Beginn erwähnt, gibt es noch einige Techniken, um Spezial-Sprites zu erzeugen. Sollten Sie zum Beispiel ein Sprite mit mehr als drei Farben wünschen, können Sie dies durch Zusammenfassen zweier Sprites erzielen.

Sprites mit 15 Farben

Damit lassen sich bis zu 15 Farben pro Sprite einsetzen. Jede der zwei beteiligten Sprite-Datenstrukturen trägt mit je 2 Bit an der Farbe des jeweiligen Spritepixels bei. Die 2 Bit zusammen ergeben damit einen 4-Bit-Wert. Es kann jedoch nur ein Sprite mit einer geraden Nummer mit dem folgenden Sprite kombiniert werden (Spritepaare: 0/1, 2/3, 4/5 und 6/7). Dabei ergeben die beiden Datenworte aus dem Sprite mit der höheren Nummer auch die beiden höchstwertigen Bits der gemeinsamen 4-Bit-Gruppe. Der Wert dieser Gruppe dient als Zeiger auf eines der fünfzehn Farbregister (Wert/Register: 1/17, 2/18, ... 15/31); der Wert Null ist wieder für die Transparenz zuständig.

Um der Videologik mitzuteilen, daß zwei Sprites miteinander verbunden werden sollen, muß das Attach-Bit (Bit 7) im zweiten Sprite-Kontrollwort des Sprites mit der ungeraden Nummer gesetzt werden.

Eine weitere Sonderform der Spritedarstellung ist das Erzeugen größerer Sprites. Dies läßt sich durch die benachbarte Positionierung von Sprites zu einem zusammenhängenden großen Sprite realisieren.

Für jedes verwendete Sprite muß eine Datenstruktur vorhanden sein. Die Zeiger auf diese Datenstruktur enthalten für Sprite 1 die Register SPR1PTH und SPR1PTL, für Sprite 2 SPR2PTH und SPR2PTL und so weiter. Das Einschalten der Sprite-DMA aktiviert alle Sprites. Deshalb müssen für die nicht benötigten Sprites Datenstrukturen ohne Bildinformationen angelegt werden.

Bei der Bewegung mehrerer Sprites ist für manche Anwendung wichtig, ob einzelne Sprites vor oder hinter anderen angezeigt werden sollen. Dies kann mit der Festlegung der Priorität eines Sprites bestimmt werden. Sind die Positionen zweier Sprites gleich, so wird dasjenige mit der höheren Priorität angezeigt. Die Entwickler des Amiga haben die Priorität entsprechend der Spritenummer festgelegt. Je kleiner die Spritenummer, desto höher die Priorität.

Selbstverständlich besitzt der Amiga auch eine Technik, mit der sich Sprites beliebig hinter oder vor einem Playfield

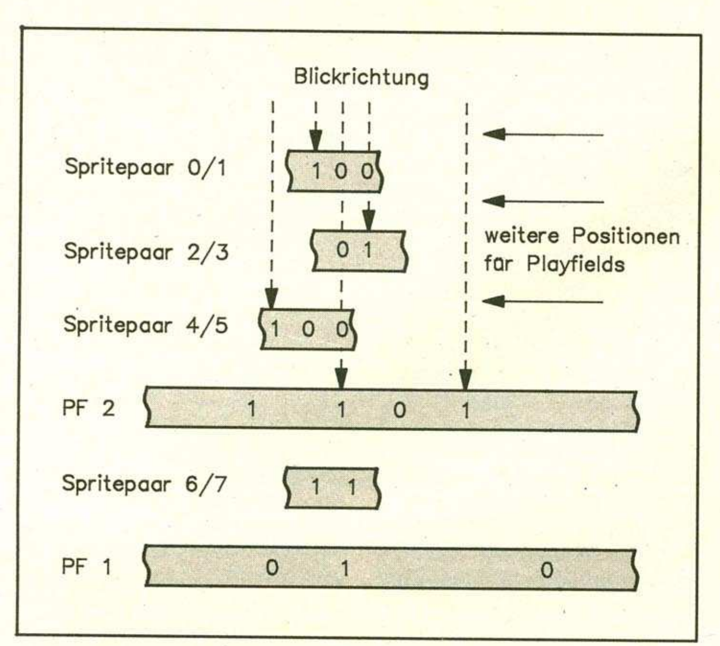


Bild 4. Die schematische Bestimmung der Priorität

bewegen lassen. Bei der Festlegung der Playfield/Sprite-Priorität werden immer zwei Sprites zusammengefaßt. Für die Zusammenfassung gilt dieselbe Regelung wie beim Sprite-Attach. Da die Priorität der Sprites untereinander nicht veränderbar ist, stellt man sich die Spritepaare am besten als Stapel mit dem Paar 0/1 obenauf vor. Blickt man von oben auf die Paare, können die darunterliegenden nur durch die transparenten Pixel gesehen werden. Die Playfields lassen sich nun zwischen den Spritepaaren hineinschieben und reihen sich so in die Prioritätenfolge ein. Dadurch sind fünf Positionen für jedes Playfield möglich (Bild 4).

Sprite-Inflation

Im BPLCON-Register legen wir die Priorität der Playfields in bezug auf die Sprites fest. Ist zum Beispiel das PF2PRI-Bit dieses Registers gesetzt, hat Playfield 2 Priorität vor Playfield 1 und erscheint vor diesem. Die 3 Bit von PF1P0 bis PF1P2 bilden einen Wert, welcher die Position des Playfields Nummer 1 zwischen den vier Sprite-Paaren bestimmt. Für Playfield 2 erfüllen PF2P0 bis PF2P2 diese Funktion.

Ist erst mal ein Sprite komplett auf dem Bildschirm abgebildet, dann hat die Sprite-Hardware in der für diesen DMA-Kanal bereitgestellten Zeit für den restlichen Aufbau des Bildschirmbildes nichts mehr zu tun. Das ist Verschwendung. Man könnte diese Zeit doch für die Darstellung eines weiteren Sprites nutzen.

Das Ende der Spritestruktur wird der Hardware durch die zwei Nullworte angezeigt. Hängen wir statt der Nullworte einfach eine weitere Datenstruktur an die erste an, so entsteht eine Mehrfach-Spritestruktur. Am Ende der zweiten Struktur müssen sich dann natürlich die beiden Nullworte befinden.

Zwischen den Spritepositionen muß mindestens eine Leerzeile liegen, damit die Kontrollworte des nachfolgenden Sprites gelesen werden können. Diese Art der Sprites nennt man virtuelle (scheinbare) oder auch VSprites, da sie ja tatsächlich nur einen DMA-Kanal und damit einen Hardware-Sprite benutzen.

Bei der Programmierung von Spielen ist es häufig sehr nützlich zu wissen, ob sich zwei Sprites berühren oder nicht. Überlappen sich die Bildpunkte zweier Sprites oder eines Sprites und eines Playfields, so wird dies als Kollision erkannt. Ausgenommen von dieser Regelung sind alle transparenten Felder, sie wären auch uninteressant. Jede erkannte Kollision wird im Kollisionsdatenregister CLXDAT gespeichert. Tabelle 1 zeigt die Zuständigkeit der Registerbits für die einzelnen Sprites und Playfields auf.

Während bei einem Sprite nur die nicht transparenten Grafikpunkte an der Kollisionserkennung beteiligt sind, ist bei den Playfields die Farbe, welche die Kollision auslöst, frei wählbar. Mit den Bits im CLXCON-Register lassen sich diese Farben einstellen (Tabelle 2). Die ENSPx-Bits des Registers geben an, ob nur ein Sprite mit gerader Nummer oder ein Spritepaar die Kollision auslöst. Die ENBPx-Bits bestimmen, welches Bitplan bei der Kollisionserkennung einbezogen beziehungsweise ausgeschlossen wird. Die MVBPx-Bits schließlich legen das Bitmuster fest, das sich in den Planes eines Playfields befinden muß, um die Kollision auszulösen. Beispiel: ENBP1-6: 111111, MVBP1-6: 100111 -> Kollision bei Kombination 100111 im Playfield; ENBP1-6: 111110, MVBP1-6: 100111 -> Kollision bei 100111 und 100110. Im zweiten Fall war Bitplane 6 ausgeschlossen und damit ist dessen Bit in der Plane irrelevant.

Damit sind wir wieder mal am Ende eines interessanten Themas. Allerdings ist immer wieder eine Funktion aufgetaucht, die noch nicht beschrieben wurde: die Steuerung der Hardware durch den Copper während des vertikalen Blankings. Damit werden wir uns in der letzten Folge dieses Kurses beschäftigen.

(Joschy Polierer/pa)

Bit Nr.	Kollision zwischen
0	Playfield 1/Playfield 2
1	Playfield 1/Sprite 0 oder 1
2	Playfield 1/Sprite 2 oder 3
3	Playfield 1/Sprite 4 oder 5
4	Playfield 1/Sprite 6 oder 7
5	Playfield 2/Sprite 0 oder 1
6	Playfield 2/Sprite 2 oder 3
7	Playfield 2/Sprite 4 oder 5
8	Playfield 2/Sprite 6 oder 7
9	Sprite 0 oder 1/Sprite 2 oder 3
10	Sprite 0 oder 1/Sprite 4 oder 5
11	Sprite 0 oder 1/Sprite 6 oder 7
12	Sprite 2 oder 3/Sprite 4 oder 5
13	Sprite 2 oder 3/Sprite 6 oder 7
14	Sprite 4 oder 5/Sprite 6 oder 7
15	Unbenutzt

Tabelle 1. Die Belegung des CLXDAT-Registers

Bit	Name	Funktion
0	MVBP1	Kollisionsmuster Plane 1
1	MVBP2	Kollisionsmuster Plane 2
2	MVBP3	Kollisionsmuster Plane 3
3	MVBP4	Kollisionsmuster Plane 4
4	MVBP5	Kollisionsmuster Plane 5
5	MVBP6	Kollisionsmuster Plane 6
6	ENBP1	Bitplane 1 mit MVBP1 vergleichen
7	ENBP2	Bitplane 2 mit MVBP2 vergleichen
8	ENBP3	Bitplane 3 mit MVBP3 vergleichen
9	ENBP4	Bitplane 4 mit MVBP4 vergleichen
10	ENBP5	Bitplane 5 mit MVBP5 vergleichen
11	ENBP6	Bitplane 6 mit MVBP6 vergleichen
12	ENSP1	Sprite 0 oder Sprite 0/1
13	ENSP3	Sprite 2 oder Sprite 2/3
14	ENSP5	Sprite 4 oder Sprite 4/5
15	ENSP7	Sprite 6 oder Sprite 6/7

Tabelle 2. Die Belegung des CLXCON-Registers

Offset hexdez	Basisadresse der Custom-Chips: ($DFF000) 14675968	
($104)260	BPLCON2	Bitplane-Kontrollregister 2
Bit6	PF2PRI	Priorität Playfield 2
Bit3-5	PF2P0-3	Priorität Playfield 2/Sprites
Bit0-2	PF1P0-3	Priorität Playfield 1/Sprites
($096)150	DMACON	DMA-Kontrollregister
($00E)014	CLXDAT	Kollisions-Datenregister
($098)152	CLXCON	Kollisions-Kontrollregister
($120)288	SPR0PTL	Spritezeiger Sprite 0 (Bit 16 bis 18)
($122)290	SPR0PTH	Spritezeiger Sprite 0 (Bit 0 bis 15)
...		
($13C)316	SPR7PTL	Spritezeiger Sprite 7 (Bit 16 bis 18)
($13E)318	SPR7PTH	Spritezeiger Sprite 7 (Bit 0 bis 15)

* Registeroffset + Basisadresse = Registeradresse

Tabelle 3. Registeradressen für die Spritekontrolle

CLI — Die zweite Haut des Amiga

Der letzte Teil des CLI-Kurses beschließt die Beschreibung der »zweiten« Benutzeroberfläche mit in älteren Amiga-DOS-Handbüchern fehlenden Befehlen und Möglichkeiten, die auf späteren Workbench-Versionen hinzugekommen sind.

Das CLI (Command-Line-Interface), das auf jeder Workbench-Diskette enthalten ist, stellt ein hilfreiches und nützliches Element zur Bedienung des Amiga dar. Mit diesem »Werkzeug« lassen sich viele (Datei-)Operationen durchführen, die von der Workbench-Ebene aus nicht erledigt werden können. Dazu gehört beispielsweise das Stellen der (wenn vorhandenen) Echtzeituhr oder die Erstellung einer Boot-Datei, die beim Start des Amiga vordefinierte Aufgaben erledigt. Doch um den richtigen Umgang mit dieser zweiten Benutzeroberfläche des Amiga zu erlernen, sind Anleitungen und Tips nötig. All dies vermittelt Ihnen dieser sieben Teile umfassende Kurs, der Ihnen die Feinheiten des CLI und der Befehle näherbringt.

Auf neueren Workbench-Versionen sind einige Befehle dazugekommen, die teilweise recht nützliche Funktionen erfüllen und die Befehlsvielfalt des mächtigen Instruments CLI erweitern. So zum Beispiel SETDATE. Dieser Befehl dient dazu, nachträglich das Erstellungsdatum und die Erstellungsuhrzeit einer Diskettendatei zu ändern. (Jede Datei auf Diskette oder im Speicher enthält zusätzliche Daten darüber, wann und um welche Uhrzeit das File erstellt wurde. Dies dient dazu, den Überblick darüber zu behalten, welche etwaigen Programmversionen älter oder neuer sind.) Angenommen, Sie besitzen ein paar Programme, bei denen Daten eingetragen sind, die nicht stimmen (etwa, weil bisher keine Echtzeituhr vorhanden war). Diese Daten sollen nun aktualisiert, also dem neuesten Stand angepaßt werden. Im Normalfall ist dies ohne einen Diskettenmonitor nicht möglich, da die Zeitdaten in bestimmten Diskettenblöcken stehen, die so ohne weiteres nicht beeinflußt werden können. SETDATE behebt dieses Manko. Der Befehl erlaubt die jederzeitige Änderung eines Erstellungsdatums und der Erstellungsuhrzeit (die Uhrzeit kann beispielsweise mit LIST angezeigt werden).

Ein Beispiel: Die Datei »Preferences« wurde mit einem falschen Datum gespeichert. Nachträglich soll das Programm auf den 27.11.1987, 12 Uhr datiert werden:

```
SETDATE Preferences
27-nov-87 12:00
```

Bei dem Befehl ist zu beachten, daß die Monate mit drei Buchstaben angegeben werden und die Schreibweise der Monate englisch ist (also MAY und nicht MAI).

Fehler werden entschlüsselt

Sicher kennen auch Sie diesen Effekt: Bei irgendeiner Dateioperation, sei es nun laden, speichern oder löschen, ist ein Fehler aufgetreten (etwa Datei nicht vorhanden) und das Betriebssystem gibt in der Menüzeile der Workbench-Oberfläche einen Fehlercode aus, der aus einer Zahl besteht. Doch was bedeuten diese Zahlen? Nicht immer ist das Amiga-DOS-Handbuch griffbereit, aus dem man die Fehlertexte ablesen kann. Hier hilft FAULT weiter. Dieser Befehl fordert den Amiga auf, den Klartext zu der jeweiligen Fehlermeldung auszugeben (jedem Fehlercode ist auch ein Fehlertext zugeordnet). Ein Beispiel: Beim Öffnen einer Datei, sei es vom CLI oder von der Workbench aus, ist ein Fehler aufgetreten, das Programm wird nicht ausgeführt. In der Menüzeile sehen wir, daß ein Fehler mit dem Code 205 aufgetreten ist. Wir möchten erfahren, welche Bedeutung der Fehlercode 205 hat, also geben wir ein:

```
FAULT 205
```

Der Amiga gibt anschließend den Klartext des Fehlers aus: »Object not found« — Datei nicht gefunden.

Sie können bis zu zehn Fehlercodes in einer Zeile angeben. Als Ergebnis erhalten Sie die Texte der Reihe nach aufgelistet. Ein Beispiel:

```
FAULT 103 202 203 204
205 213
```

Als Ergebnis erscheint:
Fault 103: Insufficient free store
Fault 202: Object in use
Fault 203: Object already exists

Bild 1. Im Normalfall erhält jeder Prozeß vom System prozentual die gleiche Prozessor-Rechenzeit zugeteilt

Teil 7

KURSÜBERSICHT

Der CLI-Kurs erläutert Ihnen in mehreren Folgen den Umgang mit der Amiga-Benutzerschnittstelle CLI (Command-Line-Interface). Am Ende haben Sie alle wichtigen Befehle kennengelernt und können die Kommandos nach eigenen Wünschen einsetzen. Die folgende Kursübersicht zeigt Ihnen, in welchen Kursteilen welche CLI-Kommandos erklärt werden:

TEIL 1: behandelte Befehle zum Anzeigen des Directories und Stellen der Uhr (DIR, LIST, CD, DATE, ENDCLI)

TEIL 2: erklärte Ihnen, wie Sie Dateien kopieren, löschen, umbenennen und Disketten formatieren können (FILENOTE, DELETE, RELABEL, RENAME, TYPE, COPY, DISKCOPY, INFO, FORMAT, INSTALL)

TEIL 3: zeigte unter anderem auf, wie Texte und Startdateien erstellt und bearbeitet werden (MAKEDIR, ED, PROMPT, WAIT, LOADWB, SETCLOCK, EXECUTE)

TEIL 4: befaßte sich mit Befehlen, die sich zum Einbau in eine Batch-Datei anbieten (ASSIGN, ECHO, FAILAT, IF, ENDIF, SAY, STACK, PROTECT, BINDDRIVERS)

TEIL 5: setzte die Erklärung von Kommandos für eine Batch-Datei fort (RUN, EDIT, JOIN, LAB, QUIT, SKIP, ADDMEM, VERSION)

TEIL 6: ging auf neue Spezialbefehle für Zusatzgeräte wie etwa Festplatten ein (WHY, NEWCLI, SORT, SEARCH, DJMOUNT, PREP, DPFORMAT, MOUNT, ADDBUFFERS, DISKCHANGE)

TEIL 7: schließt den CLI-Kurs mit den letzten Workbench-Befehlen ab (CHANGETASKPRI, DISKDOCTOR, FAULT, SETDATE, SETPREFS, STATUS)

Fault 204: Directory not found
Fault 205: Object not found
Fault 213: Disk not validated

Dieser Befehl ist sicher eine nützliche Hilfe, wenn es darum geht, Fehlermeldungen zu verstehen. Wie diese einzelnen Fehler nun zu beheben sind, können Sie einschlägiger Fachliteratur oder dem Amiga-DOS-Handbuch entnehmen. Um den Kurs nicht unnötig zu längen, verzichten wir an dieser Stelle auf das Thema »Fehlerbeseitigung«.

Erste Hilfe

Der Amiga ist in Sachen Disketten leider sehr empfindlich. Wie schnell passiert es doch, daß man eine Diskette aus einem Laufwerk nimmt, bevor der letzte Zugriff abgeschlossen ist. Schon ist es passiert: Die Diskette ist »geschädigt«, der Inhalt unrettbar (?) verloren. Dies kann sich beispielsweise dadurch äußern, daß der Amiga beanstandet, daß die Diskette nicht validiert, also nicht gültig ist oder daß sie nicht mehr gelesen werden kann. Um die Daten (so gut es geht) wieder zurückgewinnen zu können, stellt das CLI den Befehl DISKDOCTOR zur Verfügung. Er hilft, Ordnung in die Datenstrukturen zu bringen und die Daten zu retten. Allerdings restauriert DISKDOCTOR nur die Dateien, die noch genügend Informationen besitzen, die zur Wiederherstellung dienen. Das bedeutet, daß defekte Dateien verlorengehen.

Gerechte Verteilung

Im Anschluß an die Behandlung durch DISKDOCTOR sollte die gesamte Diskette kopiert werden, um mögliche Fehler auf der ursprünglichen Diskette auszuschließen. Der Aufruf von DISKDOCTOR erfolgt beispielsweise für eine in »df1« eingelegte, geschädigte Diskette folgendermaßen:

```
DISKDOCTOR df1:
```

Nachdem die anschließende Sicherheitsabfrage mit <RETURN> bestätigt wurde, beginnt Amiga-DOS, die Struktur der Diskette einzulesen und wieder zu restaurieren. Dieser Befehl sollte aber als die letzte Möglichkeit angesehen werden, defekte Disketten zu retten, da unter Umständen Daten verlorengehen können.

Wie Sie sicher wissen, ist der Amiga ein Multitasking-

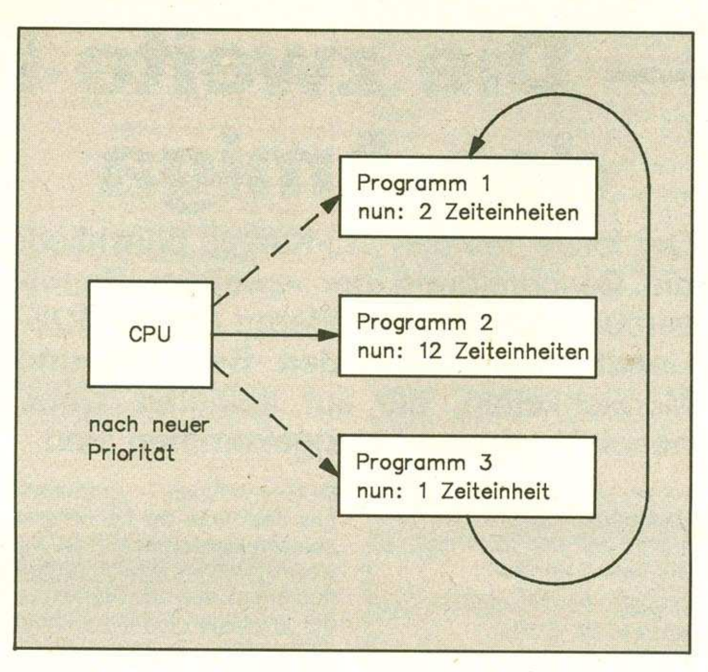


Bild 2. Nach einer Änderung erhalten rechenintensivere Programme mehr Prozessor-Rechenzeit zugeteilt

Computer. Das bedeutet, daß er scheinbar mehrere Programme gleichzeitig bearbeiten kann. Dies ist aber nur bedingt richtig. Genaugenommen wird jedes Programm eine bestimmte Zeit abgearbeitet und dann zum nächsten gesprungen (Bild 1). Das bedeutet, daß die Rechenzeit aufgeteilt wird. Je mehr Programme also »parallel« arbeiten, desto langsamer wird jeder einzelne Prozeß. Eine nützliche Möglichkeit ist es, daß der Benutzer jedem Programm eine bestimmte Rechenleistung zuteilen kann. Es ist sicher nicht sinnvoll, einem Prozeß wie etwa dem Hintergrunddrucken genausoviel Rechenzeit zur Verfügung zu stellen wie einem sehr rechenintensiven Programm. Deshalb kann mit CHANGETASKPRI die Rechenzeit eingestellt werden (Bild 2). Die neue Rechenzeit gilt allerdings nur für die Programme, die vom gleichen CLI aus aufgerufen werden, in dem auch CHANGETASKPRI verwendet wurde. Alle anderen Programme, Tasks und Prozesse, die direkt von der Workbench oder einem anderen CLI-Fenster aus aktiviert werden, laufen mit der normalen Rechengeschwindigkeit ab.

Für den Befehl können Werte zwischen -128 und 127 angegeben werden. Allerdings sollten keine zu großen Werte eingegeben werden, da sonst möglicherweise zum Betrieb des Amiga wichtige Prozesse so gebremst werden, daß das System nicht mehr sinnvoll funktioniert. Im Normalfall sind Werte zwischen -10 und 10 noch vertretbar. Ein Beispiel:

```
CHANGETASKPRI 10
```

Ab jetzt werden alle Prozesse, die der Anwender von diesem CLI aus startet, bevorzugt behandelt, also schneller abgearbeitet. Dieser Befehl ist also durchaus sinnvoll, wenn es darum geht, den Amiga effektiver arbeiten zu lassen.

Um »die Kontrolle« über das System zu behalten, kann auch der nächste Befehl dienen: STATUS. Er dient dazu, den Benutzer darüber zu informieren, welche vom CLI aus aktivierten Prozesse gerade im Hintergrund arbeiten. Wenn STATUS ohne weitere Parameter eingegeben wird, zeigt der Amiga die Nummern der CLI-Tasks sowie die Namen, mit denen sie aktiviert wurden. Angenommen, Sie haben einen offenen CLI, in dem Sie den Befehl STATUS eingeben. Als Meldung wird erscheinen:

```
Task 1: Loaded as
command: Status
```

Aber STATUS erlaubt die Eingabe von Zusatzparametern. Eines dieser Schlüsselwörter ist TCB. Wenn Sie diesen Zusatz anwenden, erscheinen Informationen über die Stackgröße zu jedem Task, der Priorität, mit der der Task bearbeitet wird sowie über die Größe die Globalvektoren. Eine Erklärung des letzten Begriffes, der wohl nur für Programmierer interessant ist, entnehmen Sie bitte dem »Technischen Amiga-DOS-Handbuch«. Eine Erklärung an dieser Stelle würde unnötigerweise zu viel Platz verschlingen. Dieser Parameter dient wohl hauptsächlich dazu, zu erfahren, mit welcher Rechenleistung, also welcher Priorität, jeder CLI-Task arbeitet.

Der zweite Parameter, der erlaubt ist, heißt CLI oder ALL. Beide Zusätze haben dieselbe Bedeutung. Laut Handbuch zeigen diese Zusätze die Sektionsnamen eines geladenen Prozesses an. Jedoch konnte dieses Resultat bei uns bisher nocht nicht beobachtet werden, so daß wohl dieser Parameter entfallen kann.

Dagegen ist der dritte und letzte Formalparameter sinnvoll: FULL. Wie der Name schon sagt, gestattet FULL, daß zu jedem Task alle möglichen Angaben ausgegeben, also daß TCB und CLI/ALL zusammen dargestellt werden. Für das obige Beispiel sähe das Resultat so aus:

```
Task 1: stk 3200, gv
150, pri 0, Loaded as
command: status
```

Soll nur ein bestimmter Prozeß angezeigt werden, ist nach STATUS die Nummer des Tasks anzugeben, also beispielsweise STATUS 1 FULL.

Zusatz für PC-Karte

Nur für PC-/AT-Karten-Besitzer ist der letzte, nur auf neuesten Workbench-Versionen enthaltene Befehl SETPREFS. Der Befehl informiert Sie darüber, mit welchen Grundeinstellungen die PC-Karte konfiguriert ist. Sie erfahren, an welchen Stellen im Speicher sich die einzelnen Emulationen und Schnittstellen befinden und welche Grafikmodi aktiviert sind. Da dies aber genausogut auch über die PC-Seite gemacht werden kann und wohl auch nicht für jeden interessant ist, wird der Befehl hier nur kurz erwähnt. Sollten Sie nähere Informationen dazu benötigen, ziehen Sie das neue Amiga-DOS-Handbuch und das PC-Handbuch zu Rate.

Mit diesem Teil ist der CLI-Kurs beendet. Sie sollten nun über genügend Wissen verfügen, um sinnvoll mit den einzelnen Befehlen umgehen zu können und auch Ihre eigenen Batch-Dateien zu schreiben, damit der Amiga in Zukunft effektiver arbeiten kann. (dm)

Unterstützende Literatur:
Das Amiga-Handbuch, Markus Breuer, Markt & Technik-Verlag, 49 Mark



C-Kurs für Anfänger

Mit den Bitfeldern und den Operatoren für Bitmanipulationen bewegten wir uns im letzten Teil auf Assemblerebene. Heute präsentiert sich die Programmiersprache C als leistungsfähige Hochsprache.

Bei der Besprechung der Operatoren unter C tauchte wiederholt der Begriff »Ausdruck« auf. Was sind eigentlich Ausdrücke? Ausdrücke stellen Konstruktionen dar, die einen bestimmten Wert liefern. Ausgehend von dieser Aussage können wir festhalten, daß eine Konstante, eine Variable oder ein Name bereits ein Ausdruck ist. Weiter können Ausdrücke Verknüpfungen von Operanden und Operatoren sein. Wenn Sie einer Variablen einen Wert zuweisen, handelt es sich schon um eine solche Verknüpfung. Diese Verknüpfung (Ausdruck) liefert einen Wert:

```
a=10
```

Es wird hier nicht nur der Variablen a der Wert 10 zugewiesen, sondern der komplette Ausdruck »a=10« liefert den Wert 10. Dieser Ausdruck ist also gleichbedeutend mit der Zahl 10. Folglich kann ein Ausdruck im Quellcode eines C-Programms überall dort stehen, wo auch ein Operand, wie zum Beispiel die Zahl 10, des gleichen Typs stehen kann:

```
printf("2 mal 5 sind
%d",10);
printf("2 mal 5 sind
%d",(2*5));
```

Die beiden Zeilen sind völlig gleichbedeutend und liefern daher auch das gleiche Ergebnis. Noch ein Beispiel:

```
b=a=10
```

»a=10« ist ein Ausdruck, der den Wert 10 liefert. Dieser Wert wird der Variablen b zugewiesen. Beide Variablen a und b erhalten also den Wert 10. Es wird ein wenig komplizierter:

```
a+2*b/4-5
```

Beachten Sie bei derartigen Formulierungen die Prioritäten der Operatoren. Legen Sie diese im Zweifelsfalle durch Klammern von Ausdrücken eindeutig fest.

```
(a+2)*b/(4-5)
```

Dieser Ausdruck liefert einen völlig anderen Wert als der vorherige, da die Prioritäten durch die runden Klammern geändert wurden. Wenn Sie sich bezüglich der Prioritäten von Operatoren nicht sicher sind, setzen Sie im Zweifelsfalle lieber zu viele Klammern, als nur eine einzige zu wenig!

Neben der Festlegung von Prioritäten steigert das Klammern von zusammengehörenden Ausdrücken die Lesbarkeit des gesamten Ausdrucks. Ausdrücke mit logischen Operatoren oder solche mit Vergleichsoperatoren liefern den booleschen Wert 0 (FALSE) oder 1 (TRUE).

```
a=(x>10)*(++b);
```

Wenn x einen Wert größer 10 hat, wird der Variablen a der um eins inkrementierte Wert von b zugewiesen. Im anderen Fall wird b ebenfalls erhöht aber a erhält den Wert 0.

Die zweite Klammer ist nicht zwingend notwendig, da der Inkrementoperator gegenüber dem Multiplikationszeichen die höhere Priorität besitzt. Die Zeile wird durch die Klammer einfach übersichtlicher.

Wenn Sie einem Ausdruck ein Semikolon folgen lassen, entsteht eine Anweisung.

```
100 a=b=c+10; /* An-
    weisung */
110 ;   /* leere An-
    weisung */
```

Eine Zeile, die nur ein Semikolon enthält, ist eine leere Anweisung. Eine solche könnte zum Beispiel in einer Warteschleife eingesetzt werden oder um eine Stelle im Quellcode besonders hervorzuheben (siehe for-Schleife ohne Rumpf). Zusammengehörende Anweisungen können in Blöcken (Bild 1) zusammengefaßt werden, welche dann vom Übersetzer wie einzelne Anweisungen behandelt werden. Ein Block wird durch geschweifte Klammern begrenzt. Innerhalb dieser Klammern können Variablen vereinbart werden, die außerhalb nicht »sichtbar«, also nur innerhalb (lokal) des Blockes bekannt sind (Listing 1).

Hinter der schließenden geschweiften Klammer eines Anweisungsblocks sollte kein Semikolon geschrieben werden, dies wäre eine leere Anweisung. Nach einem Funktionsblock (Funktionskörper) wie in der Zeile 10 darf unter keinen Umständen ein Semikolon folgen.

Teil 4

KURSÜBERSICHT

Der C-Kurs soll allen Lesern einen Einstieg in die Programmiersprache C bieten. Kenntnisse in anderen Sprachen sind nützlich, aber nicht notwendig. In den einzelnen Kursteilen finden Sie folgende Themen:

TEIL 1: Geschichte von C; das Modulkonzept; Beispielprogramme mit der Funktion »printf«, Syntaxdiagramme

TEIL 2: Einfache Datentypen, aufgeteilt nach Konstanten und Variablen

TEIL 3: Zusammengesetzte Datentypen; Zeiger, Typ-Konvertierung und Operatoren

TEIL 4: Ausdrücke und Anweisungen mit erklärenden Beispielen

TEIL 5: Aufbau von Funktionen; Parameterübergabe und Aufruf mit Programmen

TEIL 6: Der C-Präprozessor und die Bibliotheksfunktionen mit Beispielen und Übungen

Unser Programm gibt vor und nach dem Anweisungsblock als Wert der Variablen a die Zahl 10 an. Innerhalb des Blocks wird die Zahl 100 auf dem Bildschirm ausgegeben.

Eine Schachtelung von Blöcken ist erlaubt, das heißt, Sie dürfen innerhalb eines Blockes weitere Anweisungsblöcke bilden. Die Definition einer Funktion in einer anderen Funktion ist jedoch verboten.

Kontrollstrukturen unter C

Jedes Programm, ganz gleich in welcher Programmiersprache formuliert, enthält eine Liste von Anweisungen, eine Art Problembeschreibung, welche der Computer abarbeitet. Im Normalfall werden diese Anweisungen, ob sie nur aus einer Zeile oder einem ganzen Block bestehen, nacheinander (sequentiell) ausgeführt. C wäre aber keine Hochsprache, würden nicht Entscheidungs-, Wiederhol- und Sprunganweisungen zur Verfügung stehen, mit deren Hilfe man diesen Ablauf nach Belieben steuern kann.

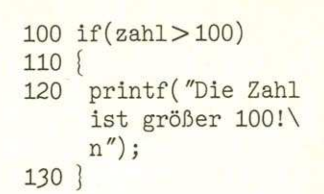
Sollen Programmteile nur unter bestimmten Bedingungen ausgeführt werden, so werden Entscheidungsanweisungen eingesetzt. Wir unterscheiden die »if-Anweisung« und die »switch-Anweisung«.

wenn ... dann ...

Bei der if-Anweisung (Bild 2) werden je nach Wert eines Ausdrucks Anweisungen ausgeführt oder übersprungen. Verbal könnte man die if-Anweisung so formulieren:

WENN der Ausdruck wahr ist, führe die Anweisung 1 aus, SONST die Anweisung 2.

Wie Sie aus dem Syntaxdiagramm (Bild 2) entnehmen können, darf der else-Teil vollständig fehlen:



```
100 if(zahl>100)
110 {
120   printf("Die Zahl
      ist größer 100!\
      n");
130 }
```

Dieses Programmfragment zeigt die ausführliche Schreibweise einer if-Anweisung. Die Anweisung, die, falls der Ausdruck TRUE liefert, ausgeführt werden soll, steht innerhalb geschweifter Klammern und stellt somit einen Anweisungsblock dar. Weitere Anweisungen können problemlos eingefügt werden. Eine if-Anweisung könnte so aussehen:

```
100 if(zahl>100)
110   printf("Die Zahl
      ist größer 100!\
      n");
```

Im zweiten Fall dieser beiden syntaktisch richtigen Formulierungen ist die Gefahr groß, daß Sie beim Einfügen von weiteren Anweisungen in den if-Zweig vergessen, die geschweiften Klammern zu setzen. Darüber hinaus ist man geneigt, Zeile 100 mit einem Semikolon abzuschließen. Dadurch wird aber die Zeile 110 immer abgearbeitet.

```
100 if(zahl>100)
110   printf("Die Zahl:
      %d\n",zahl);     /*
      Anweisung 1 */
120   printf("Die Zahl
      ist größer 100!\
      n"); /*
      Anweisung 2 */
```

Wenn die Variable größer 100 ist, wird die Anweisung 1 ausgeführt. In jedem Fall aber wird die Anweisung 2 ausgeführt. Diese gehört nicht mehr zum if-Zweig!

Im zweiten Fall ist die Anweisung zwar knapp formuliert worden, der Quellcode wird komprimiert, aber gleichzeitig nimmt dessen Lesbarkeit ab. Bei derart wichtigen Anweisungen, wie Entscheidungs- und Wiederholungsanweisungen,

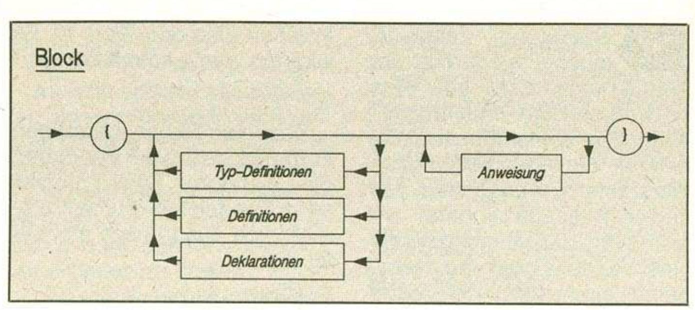


Bild 1. Zusammenfassung von zusammengehörenden Anweisungen in einem Block

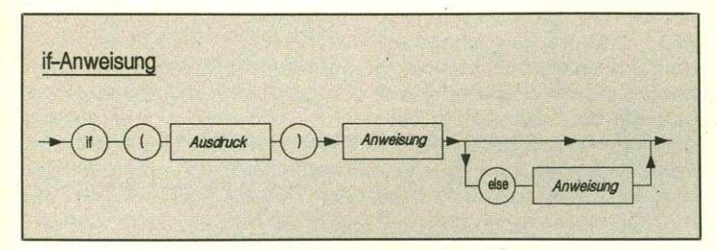


Bild 2. Die if-Anweisung steuert den Programmablauf

sollte auf ein Einrücken und auf das Binden in Blöcke nicht verzichtet werden. Schon beim Überfliegen des Quelltextes fällt sofort auf, an welchen Stellen der normale Programmablauf geändert wurde. Gleiches gilt natürlich auch für den eventuell vorhandenen else-Zweig der if-Anweisung.

```
100 if(zahl>100)
110 {
120  printf("Die Zahl
     ist größer 100!\
     n");
130 }
140 else
150 {
160  printf("Die Zahl
     ist kleiner oder
     gleich 100!\n");
170 }
180 printf("Die Zahl: %d\
    n",zahl);
```

Wenn die Variablenzahl einen größeren Wert als 100 enthält, wird die Zeile 120 ausgeführt, sonst die Zeile 160. In jedem Fall aber wird die Zeile 180 abgearbeitet. Entscheidungsanweisungen dürfen geschachtelt werden. Es ergeben sich zwei Möglichkeiten: die else-if-Folge und die if-if-Folge. Die else-if-Folge:

```
100
110 if(zahl>100){
120  printf("Die Zahl
     ist größer 100");
130 }
140 else
150  if(zahl<100)
160  {
170   printf("Die Zahl
      ist kleiner 100");
180  }
190  else
200  {
210   printf("Die Zahl
      ist genau 100");
220  }
```

Nach dem ersten else-Zweig folgt sofort eine weitere if-Anweisung. Alle dazugehörenden Zeilen werden entsprechend eingerückt. Jeder else-zweig wird dem letzten offenen if-Zweig zugeordnet. Bei der if-if-Folge empfiehlt es sich, die inneren if-Zweige einzurücken, um diese Schachtelung optisch zu verdeutlichen:

```
100 if(zahl<1000)
110 {
120  if(zahl>100)
130  {
140   printf("Die Zahl
      ist größer 100!\
      n");
150  }
160  else
170  {
180   printf("Die Zahl
      ist kleiner oder
      gleich 100!\n");
190  }
200 }
210 else
220 {
200  printf("Die Zahl
     ist größer 1000!\
     n");
210 }
```

Bild 3. Bei einer switch-Anweisung stehen verschiedene Möglichkeiten zur Auswahl

Wenn Sie solche Kaskaden auf diese Art einrücken, werden Sie auch nach längerer Zeit keine Probleme haben, Ihre Programme zu lesen. Dabei spielt es eigentlich keine Rolle, wie Sie die Zeilen einrücken oder um wie viele Zeichen. Sie sollten sich nur für eine Art entscheiden und diese dann konsequent beibehalten.

Eine weitere Möglichkeit, in Abhängigkeit eines Ausdrucks bestimmte Anweisungen ausführen zu lassen, ist die switch-Anweisung. Es handelt sich hierbei nicht um eine Entscheidung im Sinne entweder/oder, sondern es werden verschiedene Auswahlmöglichkeiten angeboten (Bild 3).

Je nach Wert des Ausdrucks wird eine der zur Auswahl stehenden Anweisungen ausgeführt. Die switch-Anweisung in Listing 2 wird nur dann ausgeführt, wenn der Wert der Variablen »zahl« zwischen null und vier liegt.

In Zeile 120 im gleichen Listing folgt dem Schlüsselwort switch in Klammern ein Ausdruck. Der Ausdruck muß einen ganzzahligen Wert liefern. Ist dies der Wert eins, so wird die Anweisung (oder Anweisungsblock) für Fall (case) 1 abgearbeitet. Die Zahl hinter case muß eine konstante int-Zahl oder ein entsprechender Ausdruck sein. Nachdem die Anweisung(en) der gefundenen case-Marke abgearbeitet sind, werden alle nachfolgenden case-Anweisungen ausgeführt, bis entweder

— das Schlüsselwort »break« gefunden wird, die switch-Anweisung bricht dann ab;
— die default-Marke wird erreicht, diese Anweisungen werden nicht berücksichtigt;
— das Ende (schließende geschweifte Klammer) der switch-Anweisung wird erreicht.

Sollte »zahl« einen Wert beinhalten, der in keinem Fall (case) berücksichtigt wurde, so wird die Anweisung hinter default ausgeführt. Die default-Marke ist optional, das heißt sie darf auch fehlen. Wenn in diesem Fall keine case-Konstante mit dem Wert des Ausdrucks hinter switch übereinstimmt, wird die switch-Anweisung ohne Reaktion verlassen.

Auch hier empfiehlt es sich, die Anweisung hinter case zu einem Block zusammenzufassen. Vor dem Ende eines jeden Blockes sollte die break-Anweisung aufgeführt werden, um ein »Durchrutschen« zur nächsten case-Marke zu verhindern. Ist dieses Durchrutschen in den folgenden case-Zweig erwünscht, so sollte dies kommentiert werden.

Die break-Anweisung hinter default in Zeile 270 ist nicht absolut notwendig, da die switch-Anweisung ohnehin nach default verlassen wird. Dennoch sollte sie angegeben werden, um deutlich anzuzeigen, daß die default-Marke an dieser Stelle beendet ist. Außerdem könnte es sein, daß Sie später noch weitere case-Alternativen anhängen möchten.

Man sollte jedoch die default-Marke immer am Ende einer switch-Anweisung belassen. Ein Beispiel für die Anwendung der Entscheidungsanweisungen finden Sie in Listing 2. Das Programm »error« gibt den Text zu einer eingegebenen AmigaDos-Fehlernummer auf dem Bildschirm aus.

In den Zeilen 4 bis 31 wird das Zeigerarray »text« initialisiert. Es enthält die Adressen der Fehlermeldungen für die Nummern 201 bis 226. Nachdem die Fehlernummer eingelesen wurde (Zeile 36) wird geprüft, ob die Fehlernummer zwischen 200 und 226 liegt. Wenn ja, wird der im Zeigerarray vereinbarte Text für diese Nummer ausgegeben. Die anderen Fehlernummern werden in der nun folgenden switch-Anweisung berücksichtigt. Die Warnungen, die der Aztec C-Compiler beim Übersetzen dieses Programms ausgibt, können ignoriert oder mit der Option -s (Aufruf: cc error.c +l -s) unterdrückt werden.

Die Wiederholungsanweisungen unter C

Dank den Wiederholungsanweisungen (Schleifen/Iterationen) können bestimmte Programmteile mehrfach abgearbeitet werden. Die Anzahl der Wiederholungen ist abhängig von einem Ausdruck (Schleifenkriterium/Anforderungen).

C verfügt über zwei Schleifenkonstruktionen, die die Prüfung dieses Ausdrucks gleich zu Beginn der Schleife durchführen und eine, die dies erst am Schleifenende vornimmt. Bei der while-Schleife (Bild 4) wird die Bedingung vor dem ersten Durchlauf der Schleife geprüft:

```
100 ende=10;
110
120 while(ende!=0)
130 {
140    ende--;
150 }
```

```
i40 main()
yR  {
rt3      int a=10;
1U       printf("a ist %d\n",a);
1U       {
dk6         int a=100;
pJ          printf("a im Block %d\n",a);
8d3      }
rT       printf("Wieder außerhalb: %d\n",a);
AfO }
```

Listing 1. Lokale Variablen in Blöcken

Der Schleifenrumpf, der hier aus einer Anweisung besteht, wird nur durchlaufen, wenn der Ausdruck, der dem Schlüsselwort while in Klammern folgt, wahr (TRUE/1) ist. Das heißt der Wert der Variablen ende muß ungleich null sein. Auch hier, wie bei der weiter oben beschriebenen if-Anweisung, müssen Sie Ihren eigenen Stil finden, wie Sie die geschweiften Klammern setzen und wie Sie den Schleifenrumpf (die Anweisungen, die wiederholt ausgeführt werden sollen) einrücken. Die folgende Zeile ist mit den Zeilen 120-150 in unserem Beispiel völlig identisch:

```
120 while(ende!=0)
    ende--;
```

Wenn die Variable ende vor der while-Anweisung einen negativen Wert hat, so kann, wie auch im folgenden Fall, das

Bedenken Sie, daß der Zeiger »string« die Adresse des ersten Zeichens der eingelesenen Zeichenkette enthält. Die Zeichen dieser Zeichenkette sind im Speicher hintereinander angeordnet, den Abschluß bildet ein Nullbyte. Die Adresse wird solange erhöht, bis dieses Nullbyte erreicht ist.

Der Ausdruck liefert mittels des Inhaltoperators den Wert des Zeichens, das an der jeweiligen Adresse gepeichert ist. Wichtig ist, daß Sie erkennen, daß der Inkrementoperator die höhere Priorität gegenüber dem Inhaltoperator hat. Somit wird nicht der Wert an der Adresse, sondern die Adresse selbst inkrementiert. Beachten Sie auch, daß wir es bei der Inkrementierung mit einer Postfix-Notation zu tun haben. Das bedeutet, daß in diesem Ausdruck zuerst der Inhalt

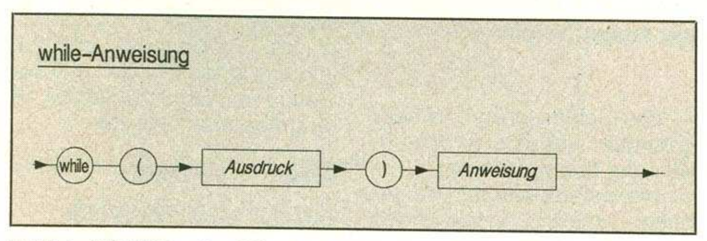


Bild 4. Mit Hilfe der Wiederholungsanweisung while können bestimmte Programmteile mehrfach abgearbeitet werden

Schleifenkriterium niemals den Wert Null liefern. Somit wird die Schleife auch nie von selbst verlassen — wir haben eine Endlosschleife formuliert.

```
100 while(1)
110 {
120    printf("Eine End-
       losschleife\n");
130 }
```

Als Beispiel für die while-Schleife dient uns das Programm strlen (Listing 3), es liefert die Anzahl der Zeichen einer Zeichenkette, die über die Tastatur eingelesen wurde. Schauen wir uns die Zeile 10 an. Solange die Schleifenbedingung einen Wert liefert, der sich von null unterscheidet, wird der Anweisungsblock (Zeile 11) ausgeführt.

der Adresse ermittelt wird. Anschließend wird diese erhöht.

Mit Komfort — die for-Schleife

Die zweite Wiederholungsanweisung, die das Schleifenkriterium vor dem ersten Schleifendurchlauf prüft, ist die for-Schleife. Diese ist der while-Schleife sehr ähnlich, auch sie wird solange durchlaufen, bis ein bestimmter Ausdruck den booleschen Wert FALSE liefert.

Der Schleifenkopf kann jedoch bis zu drei Ausdrücke (Bild 5) enthalten. Die Ausdrücke werden durch Semikolon getrennt und folgen in runden Klammern dem Schlüsselwort for.

```
z00 /*
Z78          error - Text einer AmigaDos-Fehlernummer ausgeben
6V0 */
rE  unsigned char *text[]={       /* Texte zu den Fehlernummern 20
    1-226 */
bX8          "ERROR NO DEFAULT DIR",
ft           "ERROR OBJECT IN USE",
uI           "ERROR OBJECT EXISTS",
5K           "ERROR DIR NOT FOUND",
BR           "ERROR OBJECT NOT FOUND",
hy           "ERROR BAD STREAM NAME",
Jx           "ERROR OBJECT TOO LARGE",
7i           "UNKNOWN ERROR",
up           "ERROR ACTION NOT KNOWN",
XK           "ERROR INVALID COMPONENT NAME",
q3           "ERROR INVALID LOCK",
2E           "ERROR OBJECT WRONG TYPE",
mV           "ERROR DISK NOT VALIDATED",
Y3           "ERROR DISK WRITE PROTECTED",
gA           "ERROR RENAME ACROSS DEVICES",
s0           "ERROR DIRECTORY NOT EMPTY",
Hj           "ERROR TOO MANY LEVELS",
dQ           "ERROR DEVICE NOT MOUNTED",
Bt           "ERROR SEEK ERROR",
34           "ERROR COMMENT TOO BIG",
u8           "ERROR DISK FULL",
j2           "ERROR DELETE PROTECTED",
kb           "ERROR WRITE PROTECTED",
VB           "ERROR READ PROTECTED",
3z           "ERROR NOT A DOS DISK",
cl           "ERROR NO DISK"
ok0 };
DZ  main()
Tw  {
QR4     int fehlernummer;
iZ      printf("\nBitte die Fehlernummer eingeben: > ");
Jd      scanf("%d",&fehlernummer); /* Fehlernummer lesen */
qJ      if((fehlernummer>200) && (fehlernummer<227)){
zW8         printf("%s\n",text[fehlernummer-201]);
d84     }
HC      else{
Ln8         switch(fehlernummer){ /* Text entsprechend Fehlernumm
            er ausgeben */
MKB            case 103:{
hUH                 printf("ERROR NO FREE STORE\n");
dm                  break;
JEC            }
YY             case 105:{
PgH                 printf("ERROR TASK TABLE FULL\n");
hq                  break;
nIC            }
OL             case 120:{
eMH                 printf("ERROR LINE TOO LONG\n");
lu                  break;
rMC            }
WU             case 121:{
8BH                 printf("ERROR FILE NOT OBJECT\n");
py                  break;
vQC            }
ed             case 122:{
hLH                 printf("ERROR INVALID RESIDENT LIBRARY\n");
t2                  break;
zUC            }
no             case 232:{
VgH                 printf("ERROR NO MORE ENTRIES\n");
x6                  break;
3YC            }
7U             default:{
qrH                 printf("UNKNOWN ERROR\n");
1A                  break;
6kC            } /* Ende default */
rL8         } /* Ende switch */
is4     } /* Ende else */
Xd0 } /* Ende main */
```

Listing 2. »error« demonstriert die Anwendung der Entscheidungsanweisungen. Bitte mit dem Checksummer (Seite 68) eingeben.

Im ersten Teil der for-Schleife wird einer oder, durch Kommaoperator getrennt, mehreren Variablen ein Anfangswert zugewiesen.

Es folgt das Schleifenkriterium. Liefert der Ausdruck einen Wert ungleich null, werden die Anweisungen im Schleifenrumpf wiederholt. Dies geschieht auch, wenn das Kriterium ganz fehlt, als Wert wird in diesem Fall TRUE angenommen — es handelt sich dann um eine Endlosschleife, da der Wert nicht veränderbar ist.

Der Ausdruck im letzten Teil des Schleifenkopfes wird bei jedem Ende eines Schleifendurchlaufs abgearbeitet. Hier wird in der Regel die Schleifenvariable modifiziert:

```
100 for(i=0;i<1000;i++)
110  ;
```

Sie sehen, durch die Möglichkeit der Initialisierung und der Modifizierung der Schleifenvariable kann der Schleifenrumpf völlig fehlen. Es zeugt von gutem Programmierstil, wenn man das Fehlen des Schleifenrumpfes durch ein Semikolon verdeutlicht.

Die Variable i wird vor dem ersten Durchlauf mit null initialisiert. Danach wird das Schleifenkriterium geprüft. Die Schleife wird erneut durchlaufen, wenn die Variable i einen Wert kleiner als 1000 beinhaltet. Bei jedem Schleifendurchlauf wird i um eins inkrementiert. Es erfolgt erneut die Prüfung etc. — solange i einen Wert kleiner 1000 hat.

Wenn die Schleife verlassen wird, hat i den Wert 1000, da dies der erste Wert ist, der der Bedingung nicht entspricht. Jeder der drei Ausdrücke einer for-Schleife darf fehlen — die Semikolons müssen in jedem Fall geschrieben werden:

```
100 for(;;)
110 {
120   printf("Eine Endlosschleife\n");
130 }
```

Im Gegensatz zu den uns nun bekannten Wiederholungsanweisungen prüft die do-while-Schleife erst nach dem Abarbeiten der Anweisungen im Schleifenrumpf das Schleifenkriterium. Das bedeutet, daß diese Schleife in jedem Fall einmal durchlaufen wird, auch wenn das Kriterium nicht erfüllt sein sollte.

Die do-while-Schleife (Bild 6) wird mit dem Schlüsselwort do eingeleitet. Es folgt eine Anweisung oder ein Anweisungsblock als Schleifenkörper. Das Schlüsselwort while, hinter welchem der zu prüfende Ausdruck in runden Klammern aufgeführt wird, schließt die Wiederholungsanweisung ab.

```
100 do
110 {
120    printf("Der Wert von a: %d",a--);
130 }while(a>0);
```

Auch hier möchten wir Ihnen empfehlen, den Schleifenrumpf, selbst wenn er nur aus einer Anweisung besteht, in geschweiften Klammern einzufassen. Schreiben Sie das Schlüsselwort while hinter die schließende Klammer (nicht erst in die nächste Zeile), um zu vermeiden, daß Sie es mit einer while-Schleife verwechseln. Vergessen Sie das Semikolon nicht!

Der Schleifenrumpf einer do-while-Schleife wird, wie bei den anderen Wiederholungs-

```
z00 /*
Lt5      strlen - die Länge eines Strings ermitteln
6VO */
17  main()
1U  {
a65      int i;
hc       char *string;
51       printf("Bitte geben Sie eine Zeichenkette ein! > ");
6F       scanf("%s",string);
80       while(*string++){
xvB            i++;
Ch5      }
vn       printf("Anzahl der Zeichen: %d\n",i);
EjO }
```

Listing 3. Mittels einer while-Schleife werden die Zeichen einer Zeichenkette gezählt. Bitte mit dem Checksummer (Seite 68) eingeben.

anweisungen, erneut durchlaufen, wenn die Bewertung des Schleifenkriteriums einen Wert ungleich Null ergibt.

Das Programm view (Listing 4) sollte die letzten Unklarheiten bezüglich der Wiederholungsanweisungen aus der Welt schaffen. »view« listet, ähnlich dem CLI-Kommando TYPE, ein Programmfile als Text. In diesem Programm wurden einige neue Funktionen eingesetzt. Diese sollten zuerst erläutert werden. Mittels der Funktion Open() wird ein File für die Ein- oder Ausgabe geöffnet. Die Syntax ist:

```
zeiger=Open(filename,
modus);
```

Zeiger ist hierbei ein Zeiger auf die FileHandle-Struktur. Diese Struktur ist im Headerfile dos.h in dem Verzeichnis libraries definiert. Filename ist der Name des Files, das geöffnet werden soll und modus kann entweder MODE_OLDFILE, für ein bereits existierendes File oder MODE_NEWFILE, für ein File, das neu angelegt werden soll, sein.

Die Funktion Read() wird eingesetzt, um die Daten eines Files zu lesen. Die Syntax:

```
wahre_laenge=Read(zeiger,
bereich,gewuenschte_
laenge);
```

wahre_laenge erhält die Anzahl der tatsächlich gelesenen Zeichen. Die Variable zeiger ist wieder ein Zeiger auf die FileHandle-Struktur (siehe Open()). Die gelesenen Daten werden in den Bereich »bereich« geschrieben, ein Array, das ausreichend dimensioniert sein muß. Der letzte Parameter der Read()-Funktion ist die Anzahl der Zeichen, die gelesen werden sollen. Stimmen die Anzahl der gelesenen Zeichen nicht mit der Anzahl der gewünschten Zeichen überein, so ist entweder ein Fehler passiert oder das Dateiende wurde erreicht.

Die Funktion putchar() gibt ein Zeichen auf dem Bildschirm aus:

```
putchar(zeichen);
```

Diese Formulierung ist identisch mit »printf("%c", zeichen)«. Das Gegenstück zu putchar() ist getchar(). Diese Funktion liest ein Zeichen von der Tastatur. Das Gegenstück zu Open() ist Close(). Als Parameter muß der Zeiger auf die FileHande-Struktur angegeben werden:

```
Close(zeiger);
```

Nun aber zu view, unserem neuen Programm. Nach dem Start fragt view nach einem Filenamen. Das entsprechende File wird dann als Text gelistet. Um mit Files zu hantieren ist die FileHandle-Struktur nötig, welche im Headerfile libraries/dos.h definiert ist. Dieses File wird in Zeile 4 eingebunden. Die Vereinbarung einer Variable dieser Struktur finden Sie in Zeile 7.

Das Einlesen des Filenamens mit maximal 30 Zeichen erfolgt mit Hilfe der bereits bekannten Funktion scanf().

```
 1 z00 /*
 2 bh8         view - beliebiges File als Text listen
 3 6V0 */
 4 2p  #include <libraries/dos.h>
 5 m8  main()
 6 2V  {
 7 GV8         struct FileHandle *Fhandle;
 8 T9          char filename[31], block[512];
 9 kf          int laenge, zeichen;
10 1n          printf("Bitte Filenamen eingeben: > ");
11 Zx          scanf("%30s",&filename);
12 yK          Fhandle=Open(filename,MODE_OLDFILE);
13 TW          if(Fhandle==0){
14 PaG                 printf("Kann File %s nicht öffnen!\n",filenam
                       e);
15 Fk8         }
16 to          else{
17 q5G                 printf("\fView %s\n",filename);
18 B1                  do{
19 3u0                         laenge=Read(Fhandle,block,512);
20 Z4                          for(zeichen=0; zeichen<laenge; zeich
                               en++){
21 yhW                                 putchar(block[zeichen]);
22 MrO                         }
23 SHG                 }while(laenge==512);
24 rZ                  Close(Fhandle);
25 Pu8         }
26 Qv0 }
```

Listing 4. »view« listet mit Hilfe von Wiederholungsanweisungen ein Textfile. Bitte mit dem Checksummer (Seite 68) eingeben.

Die oben beschriebene Funktion Open() dient dazu, unser File zu öffnen. Gelingt dies nicht, wenn zum Beispiel das File nicht gefunden wird, so liefert Open() den Wert Null. In diesem Fall gibt view in Zeile 14 eine Fehlermeldung aus und beendet das Programm.

Kann das File geöffnet werden, so versucht die Funktion Read() die jeweils 512 nächsten Zeichen der Datei zu lesen. Die Anzahl der tatsächlich gelesenen Zeichen übergibt Read() der Variable laenge.

putchar() gibt in Zeile 21 jedes der gelesenen Zeichen auf dem Bildschirm aus. Hier könnten Sie eine Routine vorschalten, die nur die Ausgabe der druckbaren Zeichen erlaubt. Die Steuerzeichen wie Bildschirmblitz, Umschaltung von Zeichensätzen etc. würden unterdrückt. Aber Achtung: Return ('\n') müssen Sie natürlich erlauben!

Der Schleifenrumpf wird wiederholt, wenn die Anzahl der gelesenen Zeichen mit der Anzahl der gewünschten Zeichen übereinstimmt. Ansonsten wird das File und das Programm beendet.

Abbruch von Schleifen

Manchmal ist es notwendig, eine Schleife vorzeitig abzubrechen. Die Anweisung, die dies ermöglicht, haben Sie bereits bei der Besprechung der switch-Anweisung kennengelernt: die break-Anweisung. Mit Hilfe dieser Abbruchanweisung kann jede Schleife zu einem beliebigen Zeitpunkt beendet werden:

```
100 do
110 {
120   if(j<0)
130   {
140     break;
150   }
160   else
170   {
180     j--;
190   }
200 }while(j>=0);
```

Die Variable j wird dekrementiert, bis sie kleiner als Null ist. Wenn der Wert von j bereits beim Eintritt der Schleife kleiner Null ist, wird diese gleich abgebrochen. Bei geschachtelten Schleifen wird nur die Schleife abgebrochen, in der die Abbruchanweisung steht.

Im Gegensatz zur break-Anweisung bricht die continue-Anweisung die Schleife nicht ab, sondern erzwingt nur den nächsten Durchlauf (Interation):

```
100 for(i=0; i<10; i++)
110 {
120  if(i==5)
130  {
140   printf("fünf\n");
150   continue;
160  }
170  printf("%d\n",i);
180 }
```

Die Ziffern von 0 bis 9 werden ausgegeben. Anstelle der Ziffer 5 erscheint jedoch das Wort »fünf«. Ähnlich wie Sie eine Schleife vorzeitig beenden können, ist es auch möglich eine Funktion noch vor dem Erreichen der schließenden geschwungenen Klammer zu verlassen. Die return-Anweisung tritt in Aktion:

```
100 funktion()
110 {
120  static int zaehler;
140  zaehler++;
150  if(zaehler>10)
160  {
170   printf("Diese Funk
      tion wurde schon
      mehr als 10 mal
      aufgerufen\n");
180   return();
190  }
200  zaehler++;
210  printf("Dies ist
     Aufruf Nr.: %d\n",
     zaehler);
220 }
```

In Basic wird ein Unterprogramm, das mit GOSUB angesprungen wird, mit RETURN wieder verlassen. In C erfolgt der Rücksprung aus einem Unterprogramm (Funktion) beim Erreichen der geschweiften Klammer oder beim Erreichen des Schlüsselworts return.

Die Klammer hinter return muß eigentlich nur dann geschrieben werden, wenn die Funktion Werte zurückgibt. Sie haben richtig gelesen: Man kann einer Funktion nicht nur Werte beim Aufruf übergeben, sondern sie kann auch, wie ein Ausdruck, einen Wert liefern. Dieser Returnwert wird dann in Klammern hinter das Schlüsselwort geschrieben. Eine Funktion, die einen Wert liefert, gehört der Speicherklasse dieses Wertes an. Darüber aber im nächsten Kursteil mehr.

Ergänzen Sie Zeile 180 in unserer Funktion wie folgt:

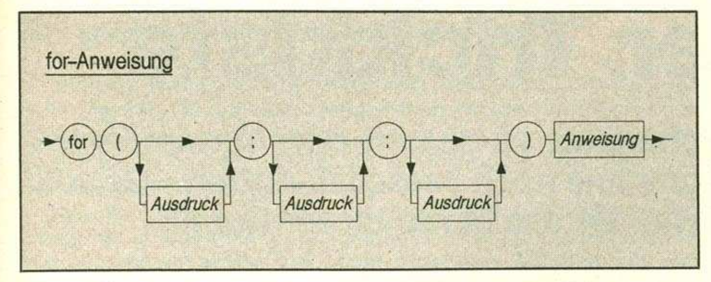


Bild 5. Die for-Schleife ist eine erweiterte while-Schleife mit vielen verschiedenen Möglichkeiten

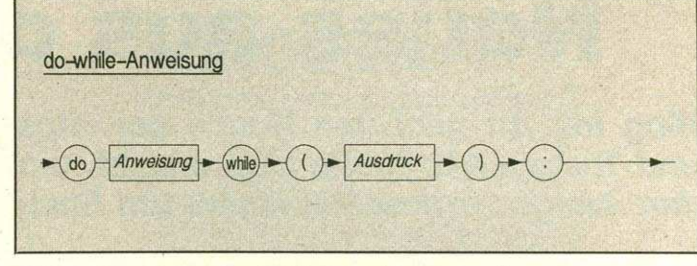


Bild 6. Eine do-while-Schleife prüft die Bedingung erst am Ende eines mindestens einmal erfolgten Durchlaufs

```
180     return(zaehler);
```

Die Funktion kehrt nun mit dem Wert der Variablen zaehler zurück. Dieser Wert kann zum Beispiel beim Aufruf der Funktion einer Variablen übergeben werden:

```
a=funktion();
```

Der Variablen a wird der Returnwert der Funktion funktion() zugewiesen. Sie sehen, C bietet eine Vielzahl von Möglichkeiten den Programmablauf beliebig zu steuern. Eine Anweisung zu diesem Thema fehlt aber noch: die goto-Anweisung. Wir werden sie hier nur der Vollständigkeit halber erwähnen, denn bei der strukturierten Programmierung, wie sie C erlaubt, kommt man sehr leicht ohne sie aus.

Mit Hilfe der goto-Anweisung kann man innerhalb einer Funktion beliebige Sprungmarken (Labels) anspringen. Diese Sprungmarken sind Anweisungen, denen ein Name mit Doppelpunkt vorangestellt wird:

```
100 LOOP: i++;
110 goto LOOP;
```

In der nächsten Ausgabe unseres C-Kurses werden wir uns intensiv mit Funktionen befassen. Auch dann werden Sie wieder einige interessante Beispielprogramme finden. Bis dahin sollen Sie aber noch ein bißchen üben, um mit C vertraut zu werden. Es geht um das Programm »view«:

1) Modifizieren Sie das Programm so, daß auch Programmfiles problemlos gelistet werden können. Sie müssen hierzu die Steueranweisungen unterdrücken. Eine ASCII-Tabelle finden Sie im Anhang Ihres Basic-Handbuchs.

3) Immer wenn 20 Zeilen (1 CLI-Fenster) ausgegeben wurden, soll der Benutzer zwischen Programmabbruch und Weiterlisten wählen können. Vor dem Weiterlisten soll der Bildschirm gelöscht werden.

4) Schreiben Sie ein Programm, das die Anzahl der Zeilen eines Files bekanntgibt.

5) Eine Aufgabe für Spezialisten: Ein Programm soll die Anzahl der Blätter, die nötig sind, um ein beliebiges File auszudrucken, auf dem Bildschirm ausgeben. Schauen Sie sich hierzu, neben view und Ihrer Lösung der heutigen vierten Aufgabe, das Programm »Prefs« aus der letzten Ausgabe des C-Kurses an.

Darin versteckt sich nämlich eine Information, die Sie benötigen, um die Berechnung richtig durchführen zu können. Aber Sie ahnen sicher schon, welchen Wert Sie brauchen.

(Arno Gölzer/rb)

Neues aus der Trickkiste

Ring frei zur nächsten Runde der »Tips und Tricks«. Alle Kniffe sind erlaubt, um den Amiga zu beschleunigen, um Speicherplatz zu sparen oder Anwenderprogramme noch besser zu nutzen. »Gewußt wie« ist das Motto dieser Rubrik.

Wieder öffnet sich die Trickkiste zum Amiga. Sie ist gefüllt mit Juwelen und Diamanten für jeden Amiga-Besitzer, vom Einsteiger bis zum Profi. Unter den Schätzen finden Sie diesmal wieder Informationen zum CLI, Ratschläge zur Programmierung, Hilfen für Anwenderprogramme, Hardware-Basteleien und noch manches mehr.

Color-Modus für PC-Erweiterung

Manch ein Besitzer der PC/XT-Erweiterung für den Amiga 2000 ist im Color-Modus einer unfreundlichen Prozedur ausgesetzt:

Jedesmal muß er blind den Befehl MODE CO80 zum Einstellen des Farbcodes eingeben. War die Eingabe richtig? Auf dem Bildschirm ist nichts zu sehen. Da lohnt es sich, die Einstellung der Farbe direkt vom Computer vornehmen zu lassen.

MS-DOS sucht beim Starten des Systems nach einer Stapelverarbeitungsdatei. Es arbeitet die darin abgelegten Befehle automatisch ab. Diese Datei namens »AUTOEXEC.BAT« können Sie mit Hilfe des Zeileneditors EDLIN verändern. Geben Sie nacheinander folgendes ein:

```
EDLIN AUTOEXEC.BAT <Return>
L <Return>
I <Return>
MODE CO80 <CTRL-C>
E <Return>
```

Jetzt wird der entscheidende Befehl zur Einstellung der Farbe im Color-Modus automatisch durchgeführt. Dem schnellen und bequemen Benutzen des PC-COLOR-Modus steht nichts mehr im Wege.

Wichtig: Falls sich EDLIN nicht auf der MS-DOS Diskette befindet, sollten Sie diesen Zeileneditor von der mitgelieferten GW-Basic-Diskette kopieren. (R.Giebichenstein/ub)

Oh Gemini: Treiber für alte Drucker

Mancher Besitzer eines älteren Epson-kompatiblen Druckers, zum Beispiel des Star Gemini 10x wird beim Ausdruck von Amiga-Grafiken eine böse Überraschung erleben: Das Ergebnis erinnert eher an einen Zebrastreifen als an eine Hardcopy. Die Ursache hierfür liegt in der Steuersequenz des Epson-Druckertreibers für einen Zeilenvorschub um 24/216 Zoll. Einige Drucker führen statt dessen einen Vorschub um 24/144 Zoll aus. Der Zeilenabstand wird zu groß. Die Betroffenen können das Problem beseitigen, indem sie statt der 24/216-Zoll-Steuersequenz die für 8/72 Zoll »$1B, $41,$08« verwenden. Folgendes Basic-Programm nimmt alle Änderungen vor und speichert den modifizierten Treiber unter dem Namen »Epson_k«.

```
;Epson-Patch für Gemini-Drucker
CHDIR "DEVS:printers"
Such$=CHR$(27)+CHR$(51)+CHR$(24)
OPEN "epson" FOR INPUT AS 1
Driver$=INPUT$(LOF(1),1):CLOSE 1
p=INSTR(Driver$,Such$)
IF p=0 THEN PRINT "Falscher Epson- Treiber":END
MID$(Driver$,p+1)=CHR$(65)+CHR$(8)
OPEN "Epson_K" FOR OUTPUT AS 1
PRINT#1, Driver$;
CLOSE 1:KILL"epson_K.info;
```

Getestet wurde das Programm mit dem Epson-Druckertreiber von der A500-Workbench 1.2 Version 33.56. Gerade vom Epson-Druckertreiber sind unterschiedliche Versionen im Umlauf. (Alexander Hoernik/ub)

(x)yz gelöst: Die Gilde der Diebe

»The Guild Of Thieves« ist ein Abenteuerspiel, welches direkt gebootet wird. Sie können es nicht von der Workbench aus laden. Leider läuft das Programm nur mit dem amerikanischen Tastaturtreiber. Die Vertauschung von <y> und <z> ist besonders störend. Doch dies läßt sich vermeiden: Erstellen Sie zunächst eine Sicherheitskopie der Originaldiskette. Fügen Sie in der »Startup-Sequence« der Kopie diese Zeile ein:

```
Setmap d
```

Den Befehl SETMAP und den Treiber für die deutsche Tastatur finden Sie auf jeder Workbench-Diskette. Kopieren Sie SETMAP aus dem Verzeichnis »systems« in den C-Ordner der Boot-Diskette. Legen Sie ebenfalls auf der Boot-Diskette mit MAKEDIR ein neues Dateiverzeichnis mit dem Namen »devs/keymaps« an:

```
MAKEDIR devs/keymaps
```

In dieses kopieren Sie die Datei »d« aus dem gleichnamigen Verzeichnis der Workbench-Diskette. Wenn Sie nun den Amiga mit der neu hergestellten Diskette starten, wird der deutsche Tastaturtreiber geladen. (Sven Friedrich/ub)

Disketten retten durch Kopieren

Wer hat sich noch nicht über die Meldung »Read/Write Error« geärgert und die defekte Diskette mit ein paar Tränen aufgegeben? Alle Daten waren verloren, auch wenn nur kleine Bezirke auf der Diskette zerstört waren. Dies muß nicht sein. Wer ein Kopierprogramm besitzt, das Lesefehler ignoriert, zum Beispiel das »SOLOCOPY« aus der 68000er, kann einen Teil seiner Diskette eventuell noch retten. Kopieren Sie die gesamte Diskette auf eine andere. Das Kopierprogramm wird die Lesefehler — meist Nummer 21 oder 25 — nicht beachten und jede lesbare Information auf die zweite Diskette übertragen. Danach sind diese Werte auf der neuen Diskette wieder lesbar. ASCII-Dateien lassen sich auf diese Art gut restaurieren. Das Verfahren scheitert meist bei IFF-Files, da durch einen Fehler, zum Beispiel ein fehlendes Byte, die Struktur des gesamten Files zerstört wird. (Sven Stillich/ub)

Wer hat den Amiga entwickelt?

Kennen Sie die Entwickler des Amiga. Interessenten können deren Namen ermitteln. Aktivieren Sie die Workbench. Drücken Sie gleichzeitig beide <Shift>-Tasten sowie beide <ALT>-Tasten — wenn Sie jetzt auch noch eine der 10 Funktionstasten betätigen können, erscheinen am oberen Bildschirmrand die gesuchten Namen. Die Liste der Namen reicht von — halt, schauen Sie am besten selbst nach. Besonders die Taste F10 ist recht amüsant. (Marco Trauner/ub)

Amiga-Tuning ohne Turbo

Das Darstellen von vielfarbigen Screens bremst den Amiga. Zum Beispiel der Ausdruck eines HiRes-Bildes mit dem Malprogramm Butcher dauert vergleichsweise lange. Eine Beschleunigung läßt sich durch das »Nach-vorne-klicken« des Workbench-Screens erreichen. Noch besser, Sie ziehen alle sichtbaren Screens mit der Maus so weit nach unten, daß der Amiga vollständig von der Bilddarstellung befreit ist. Auch zeitaufwendige Mandelbrotprogramme lassen sich beschleunigen. (Michael Holin/ub)

So geht's: Zeilen einrücken

Während des Programmierens mit Amiga-Basic kommt es beim Einrücken oder Hochziehen einer Programmzeile mit der Backspace-Taste zu der Fehlermeldung »line Buffer overflow«. Im Basic-Handbuch steht hierzu: »Es wird versucht, eine Zeile mit mehr als 255 Zeichen einzugeben.«

Dies ist in diesem Beispiel aber gar nicht der Fall. Nehmen Sie vorher mit <rechte Amiga>-Taste und <x> ein Leerzeichen in den Korrekturspeicher auf. Jetzt tritt beim Arbeiten mit der <Backspace>-Taste kein Fehler mehr auf.

(Birger Fahning/ub)

Reset per Programm auslösen

Der Amiga läßt sich rein softwaremäßig in den Grundzustand versetzen. In Basic funktioniert dies recht einfach:

```
Adresse=16515072 ; Beginn des Amiga-ROM
CALL Adresse
```

In Assembler lautet der entsprechende Befehl:

```
jmp $fc0000
```

In beiden Fällen blinkt zunächst die Power-LED. Kurz darauf erscheint auf dem Bildschirm das Workbench-Symbol. Ist <CapsLock> aktiv, während Sie auf diese Art einen Reset auslösen, so leuchtet die Leuchtdiode der Taste weiter. <CapsLock> muß vor dem nächsten Gebrauch zweimal gedrückt werden.

(Alexander Hoernik/ub)

Welcher Monitor ist der richtige?

Wer mit dem Amiga arbeiten will, sollte sich überlegen, ob er dazu wirklich einen RGB-Monitor benötigt. Monochrom-Bildschirme lassen zwar nicht die Darstellung der Farbenpracht des Amiga zu, aber in der Textdarstellung sind selbst mittelmäßige monochrome besser als farbige Monitore. Die Bildschirmfarben Orange, Amber oder Bernstein liefern ein flimmerfreies Bild. Grüne Monitore, die lange nachleuchten, meistern sogar den Interlace-Modus ohne zu flackern. Allerdings ziehen bewegte Objekte auf dem Bildschirm einen Kometenschweif hinter sich her. Für Spiele sind solche Monitore weniger geeignet.

Der Anschluß eines Monochrom-Monitors ist beim Amiga 500 einfach. Der Computer verfügt über einen passenden Ausgang mit der Kennzeichnung »Mono«. Sie müssen diese Buchse mit einem einfachen Cinchkabel mit dem Eingang des Monitors verbinden. Beim Amiga 1000 und Amiga 2000 benötigen Sie einen Adapter (erhältlich in gutsortierten Fachgeschäften). An den Composite-Videoausgang des Amiga 1000 sollten Sie keinen Monochrom-Monitor anschließen. Der Monitor würde den Farbhilfsträger des Composite-Signals als Helligkeitssignal interpretieren. Dies führt meist zu einem verschmierten Bild.

(Alexander Hoernik/ub)

»Flushlibs« löscht Bibliotheken

Teile des Betriebssystems, die der Amiga von der Workbench nachlädt, die sogenannten »Libraries«, verschwinden nach ihrem Gebrauch nicht sofort aus dem Speicher. Der Vorteil: Bei einem weiteren Aufruf von Routinen dieser Bibliotheken müssen sie nicht noch einmal von der Diskette geladen werden. Aber das kostet wertvollen Speicherplatz. Auf der Workbench gibt es einen Menüpunkt, der das Löschen von Libraries im Speicher bewirkt. Dieser Menüteil ist jedoch in der Regel inaktiv. Sie müssen ihn vorher selbst aktivieren. Ändern Sie in der »Startup-Sequence« den Aufruf der Workbench:

```
LOADWB -Debug
```

Sie können den Befehl auch direkt im CLI eingeben. Fahren Sie nun mit gedrückter rechter Maustaste auf der Menüleiste an den rechten Rand. Dort erscheint neben dem »Spezial-Menü« ein weiterer Menüpunkt. Flushlibs gibt den Speicherplatz aller im RAM residenten Libraries frei.

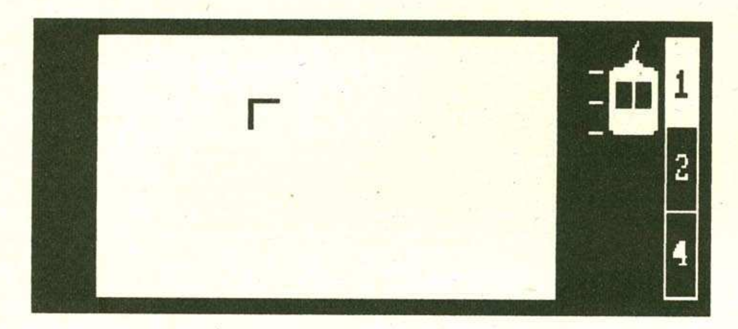
(Alexander Hoernik/ub)

Ein riesiges Basic-Fenster

Wie schon im Leserforum im AMIGA-Magazin, Ausgabe 11, erwähnt, ist das normale Basic-Fenster des Amiga nur maximal 187 Punkte groß. Möchten Sie ein größeres Window? Möchten Sie den gesamten PAL-Bildschirm nutzen? Hier ist die Lösung:

Als erstes müssen Sie einen neuen Screen für das Fenster öffnen. Die im Basic-Handbuch angegebenen Höchstgrenzen für den SCREEN-Befehl sind falsch. Selbst auf einem Standard-Amiga lassen sie sich geringfügig überschreiten. Um ein großes Fenster zu programmieren, muß ein Trick angewendet werden:

Verschieben Sie erst mit den »Preferences« den Bildschirm nach links oben. Dabei müssen Sie in Kauf nehmen, daß der Text in der Menüleiste bei manchen Monitoren schlechter zu lesen ist.



Die im Bild gezeigte Einstellung stellt einen guten Kompromiß dar. Geben Sie anschließend in Basic dieses Listing ein:

```
SCREEN 1,673,269,1,2
WINDOW 1,"Grafik",(0,0)-(664,255),10,1
WINDOW OUTPUT 1
FOR y=1 to 255
  w=w+1
  FOR x=1 TO w
    PSET (664*RND(1),y)
  NEXT x
NEXT y
```

Wenn Sie dieses Programm starten, so sehen Sie eine Grafik-Demonstration mit 664*255 genutzten Punkten — diese Zahl berücksichtigt noch nicht einmal den Rand des Fensters.

(Andreas Imhoff/ub)

Hilfe für den Floppyspeeder

»Facc« und »Facc II« sind nützliche Programme, die bei vielen Diskettenzugriffen Zeit einsparen. Ein Nachteil ist, daß der Cache-Speicher — anders als beim Befehl ADDBUFFERS — von vornherein auf 256 KByte eingestellt ist. Wem dies zuviel ist, der kann die Puffergröße zwar jederzeit ändern, doch ist diese Arbeit vor jeder Benutzung des »Facc« recht lästig. Der folgende Patch schafft in solchen Fällen Abhilfe:

— Facc: Laden Sie mit dem Filemonitor »NewZap« oder »FileZap« (Public Domain) den Sektor 14. An den Cursorpositionen 126 und 127 ist die Puffergröße gespeichert ($0100 = 256). Ändern Sie den Wert nach Ihren Wünschen.

— Facc II: Mit derselben Methode können Sie auch Facc II verändern, jedoch liegt die gesuchte Variable hier in Sektor 1 an der Position $15C und $15D. Arbeiten Sie auf jeden Fall mit einer Sicherheitskopie des Programms. (Michael Holin/ub)

Besseres TYPE: Mehr, mehr, mehr

Wer ASCII-Dateien betrachten möchte, kann hierzu an Stelle von TYPE auch den Befehl MORE benutzen. Das Programm MORE befindet sich auf der ExtraD-Diskette im Ordner »Tools«. Es kann ohne weiteres ins Verzeichnis C einer CLI-Diskette kopiert werden. MORE gibt Texte bildschirmweise aus. Drücken Sie nach Aufruf dieses Utilities einfach <H> für Hilfe und Sie erhalten eine Gebrauchsanweisung des Programms.

(Alexander Haderer/ub)

Lieber schlafen als lange warten

Das Basic-Statement SLEEP erscheint nur selten in Programmen. Es könnte in der Kombination mit der Maus nützlich angewendet werden. Wenn ein Programm warten soll, bis Sie die Maustaste anklicken, ist diese Subroutine geeignet:

```
Warte_auf_Klick1:
   WHILE ( MOUSE(0)=0)
   WEND
RETURN
```

Diese Routine hat den Nachteil, daß der Prozessor viel Zeit verschwendet. Immer wieder durchläuft er die WHILE-Schleife. Die Alternative hierzu ist eine Routine, die mit SLEEP arbeitet:

```
Warte_auf_Klick2:
 ON MOUSE GOSUB Geklickt
 MOUSE ON
 Nochwach = 1
 WHILE (NochWach)
   SLEEP
 WEND
RETURN
Geklickt:
 Nochwach = 0 : RETURN
```

Diese Mausabfrage ist besser, da der Prozessor andere Aufgaben wahrnehmen kann. Der Amiga ist eine Multitasking-Maschine. Wenn ein Task nur auf die Maus wartet, so sollte er auch keine wertvolle Prozessorzeit beanspruchen. Erst der Druck einer Maustaste im zum Task gehörenden Fenster reißt die Subroutine aus dem Schlaf. (Tis Veugen/ub)

Von Zaubertränken und Fallen

Zwei Tips für alle Garrison-Liebhaber: Bei diesem Spiel finden Sie in den Räumen zahlreiche Zaubertränke (Potions) aber auch Fallen (Traps). An der Farbe können Sie deren Wirkung erkennen.

Potions:	Wirkung
weiß	Magie +1
gelb	Nahkampf +1
rosa	Schutz gegen Treffer +1
grün	Schußgeschwindigkeit +1
orange	Schußwirkung +1
Fallen:	
gelb	Lähmung
rosa	Punktabzug
rot	Bombe
blau	Kraftverlust
grün	Charaktermerkmal -1

Ein weiterer Tip betrifft die zahlreichen Sensenmänner, die in manchen Räumen umherlaufen. Diese lassen sich nur schwer bekämpfen und richten großen Schaden an. Sie können sich dieser Störenfriede mit Hilfe einer Zauberrolle, einer »Scroll«, entledigen. Eine Scroll wird aktiviert, wenn Sie <Shift> drücken. Sie können die Bösewichter auch in den Kugelhagel anderer Feinde locken oder mit Hilfe der Teleports vernichten. Probieren Sie es aus, es ist nicht schwer. (Jörg Dieter Bendrich/ub)

Starkes Duo: CLI und Workbench

Geht es Ihnen auch so: Gerade sind Sie mitten drin im Arbeiten auf der Workbench, plötzlich benötigen Sie einen CLI-Befehl. Als erstes müssen Sie nun die System-Schublade öffnen, das CLI laden und auch noch das Fenster vergrößern. Das geht einfacher. Fügen Sie einen kurzen Befehl in der »Startup-Sequence« ein:

```
NEWCLI con: 540/200/60/50/CLI-Fenster
```

Binden Sie den Befehl vor dem Befehl LOADWB ein. Nach dem Booten finden Sie auf der Workbench unten rechts ein kleines CLI-Fenster. Es steht Ihnen bei Bedarf sofort zur Verfügung.

(Peter Riedlberger/ub)

Sound und Wave: Das ist wichtig

1. Wenn mit WAVE eine Wellenform definiert wird, so werden nur die ersten 256 Werte des übergebenen Integerfeldes für die Tonerzeugung verwendet:

```
RESTORE: DIM a%(400)
FOR i = 0 TO 255:
  READ a%(i)
NEXT
WAVE 0,a%
```

Dieses Programm ist identisch zu dem folgenden, soweit gleiche DATA-Tabellen übergeben werden:

```
RESTORE: DIM a%(400)
FOR i = 0 TO 400:
  READ a%(i)
NEXT
WAVE 0,a%
```

Das bedeutet, daß die Werte a%(256-400) nicht anerkannt werden.

2. Tritt bei SOUND ein Knacken oder Knarren auf, so liegt dies eventuell an einer zu kurzen Abspielzeit. Der Amiga hat einen Ton zu Ende gespielt und noch keine Anweisung für den nächsten. Vergrößern Sie in diesem Fall die Tondauer des SOUND-Befehls (2. Parameter). (Alexander Haderer/ub)

Hardcopy-Routine im Programm

Auf der Diskette »Amiga Extras 1.2« befindet sich in der Schublade »BasicDemos« das Programm »ScreenPrint«. Dieses können Sie in Ihren eigenen Basic-Programmen verwenden.

— Entfernen Sie zunächst den oberen Programmteil bis zur Zeile »Screendump:«.

— Löschen Sie auch alle REM-Zeilen; das spart Speicherplatz. Lesen Sie die Kommentare in den REM-Zeilen jedoch einmal durch. Hier finden Sie wertvolle Erläuterungen zu »Screendump«.

— Speichern Sie das Listing mit »SAVE "<name>",a «.

— Mit »MERGE "<name>"« läßt sich die Hardcopy-Routine in jedes Basic-Programm einbinden.

Der Aufruf erfolgt entweder über ein Pull-Down-Menü oder mit Hilfe des Befehls »GOSUB Screendump«. Der Amiga beginnt danach sofort mit dem Ausdruck des gerade sichtbaren Bildschirms. Wichtig: Wenn Sie »ScreenPrint« auf eine andere Diskette kopieren, muß sich auf dieser die Datei »exec.bmap« befinden. (Ralf Thomas Barding/ub)

Zeit einstellen leicht gemacht

Wem es zu lästig ist, nach dem Booten die Preferences zu laden, um das Datum einzustellen, ist mit der Public Domain-Diskette »Faug Hot Mix Nummer 34« gut beraten. Auf ihr befindet sich das Programm »TimeSet«. Dies erlaubt es Ihnen, die Zeit komfortabel einzugeben. Es ist von Vorteil, dieses Utility mit RUN in die »Startup-Sequence« einzubinden:

```
RUN TimeSet
```

Nun können Sie das Datum sogar einstellen, während der Amiga die weiteren Befehle der Startsequenz ausführt. (Peter Riedelberger/ub)

Basic steuert die Hardware

In der AMIGA 8/9 steht in der Rubrik »Tips und Tricks«, wie sich die Power-LED mit Hilfe eines Assemblerprogramms ein- und ausschalten läßt. Auch Basicprogrammierer können auf die Hardware zugreifen:

```
POKE 12574721,254 schaltet die LED aus
POKE 12574721,252 schaltet die LED ein
```

Lassen Sie in Ihren Programmen die Leuchtdiode ein paarmal blinken. Verblüffen Sie mit diesem Effekt Ihre Bekannten. (Daniel Swertz/ub)

Es geht auch ohne Mäuse

Gehören Sie zu den Anwendern, die die Maus auch mal beiseite legen und nur mit dem CLI arbeiten? Insbesondere C- und Assemblerprogrammierer benötigen keine Workbench-Umgebung. Dann werden Sie sicherlich eine Diskette anlegen, bei der das AmigaDOS-Fenster nach dem Booten gar nicht erst geschlossen wird. Am einfachsten geht dies, indem Sie den Befehl LOADWB in der »Startup-Sequence« entfernen. Dabei entsteht das Problem, daß das Fenster nach dem Start mit der Maus auf die volle Größe gebracht werden muß, um die PAL-Auflösung zu nutzen. Doch wo ist die Maus? Wieder müssen Sie dieselbe aus der Schreibtischschublade hervorkramen. Aber halt, es geht auch einfacher. Dieses C-Programm, in die »Startup-Sequence« eingebunden, schafft Abhilfe:

```
#include <exec/types.h>
#include <intuition/intuition.h>
struct IntuitionBase *IntuitionBase;
main()
{
IntuitionBase = (struct IntuitionBase*)
     OpenLibrary ("intuition.library",0);
if (IntuitionBase == 0) exit (0);
  SizeWindow (IntuitionBase -> ActiveWindow, 0,
  56);
  CloseLibrary (IntuitionBase);
}
```

In den ersten beiden Zeilen werden die benötigten Include-Files eingebunden. Der Zeiger auf die IntuitionBase wird eingerichtet und in main() die Intuition-Bibliothek geöffnet. Die Funktion »SizeWindow« verändert die Größe eines Windows:

```
SizeWindow (Zeiger auf Fensterstruktur, dy, dx)
```

In diesem Fall wird das aktuelle Fenster, also das CLI-Window um 56 Pixel in der Vertikalen vergrößert. Zum Schluß wird die Library geschlossen und zurück zum CLI gesprungen. Eine einfache, aber wirkungsvolle Sache. (Dirk Brammerts/ub)

Joystickabfrage in Assembler

Für Programmierer von Actionspielen ist die Verwendung von Assembler eine Alternative zu Basic oder C, um schnelle Programme zu erzeugen. Um von einem Maschinenprogramm den Joystick abzufragen, gibt es prinzipiell zwei Verfahren:

1. Sie können das »Gameport-Device« verwenden. Der Haken an der Sache: Ihr Programm wird aufgebläht und langsam.

2. Als Alternative bleibt, den Joystickport direkt abzufragen. Greifen Sie mittels weniger Assemblerbefehle direkt auf die Hardwareregister zu. Das folgende Programm kehrt nach seinem Aufruf mit einem Wert in d0 für den Joystick in Port 0 zurück. Den entsprechenden Wert für Port 1 finden Sie in d1:

```
 JOY0DAT =$DFF00A
 JOY1DAT =$DFF00C
 JOYTEST =$DFF036
LeseJoystick:
 clr.w JOYTEST      ; Löschen der Mausregister
 move.w JOY0DAT,d0  ; JoyPort 0 lesen
 move.w Joy1DAT,d1  ; JoyPort 1 lesen
 rts
```

In den Registern d0 und d1 stehen je nach Stellung des angeschlossenen Joysticks diese Werte:

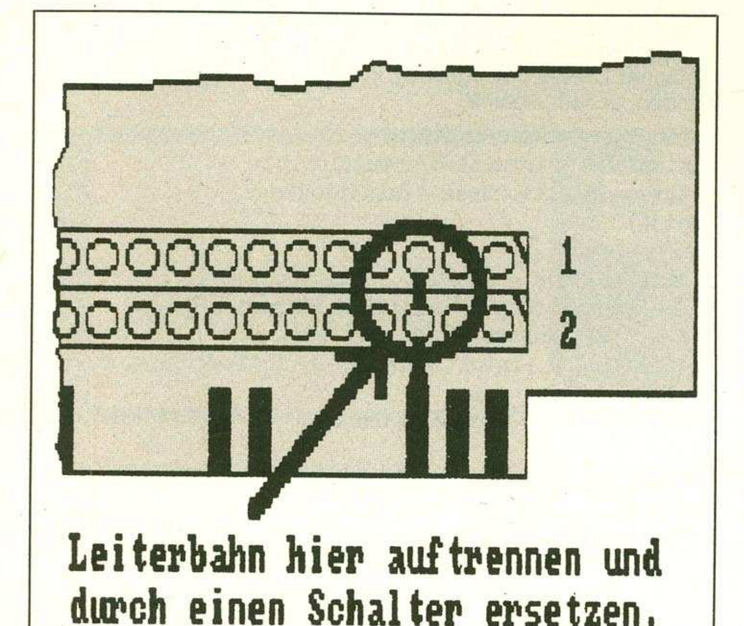
$0200 links
$0300 links oben
$0201 links unten
$0100 oben
$0000 Mitte
$0001 unten
$0102 rechts oben
$0002 rechts
$0003 rechts unten

Einfacher und schneller geht es nicht.

(Jörg Schmidt/ub)

NoFastMem für den Amiga 2000

In der AMIGA 8/9 finden Sie in der Rubrik Tips und Tricks eine Umbauanleitung, um die Speichererweiterung beim Amiga 2000 abzuschalten. Hier ist eine einfachere Version. Da auf der Erweiterungskarte im Amiga 2000 keine Uhr berücksichtigt zu werden braucht, kann die Abschaltung des Speichers durch eine Unterbrechung der Spannungsversorgung erfolgen. Nehmen Sie hierzu die Platine; die Bauteileseite auf Sie zugerichtet.



Zwischen Pin 5 und 6 müssen Sie trennen

Trennen Sie auf der Erweiterungskarte die in der obigen Zeichnung markierte Leiterbahn zwischen den Kontaktfahnen an Pin 5 und 6 durch. Diese Trennstelle müssen Sie nun mit einem Schalter überbrücken. Den Schalter können Sie außen am Amiga befestigen. Achtung: Vor jedem Umschalten muß der Amiga ausgeschaltet werden. Bedenken Sie bitte, daß Sie durch Arbeiten an der Hardware jegliche Garantieansprüche verlieren. Für eventuelle Schäden, die beim Umbau entstehen, wird keine Haftung übernommen. (Dieter Schlingmann/ub)

Die Lösung für den neuen Amiga

Im AMIGA-Magazin 12/87 fragte Herr de Heuvel nach den Unterschieden zwischen dem Amiga A2000 und B2000.

Differenzen bestehen in der Hardware. Verschiedene Bauteile des alten Amiga 2000 sind durch »FAT-Agnus« und »Garry« ersetzt worden. Der B2000 enthält im Gegensatz zum A2000 1 MByte auf der Hauptplatine. Die oberen 512 KByte sind durch einen Jumper abschaltbar. Damit wird der Einbau eines Schalters, um das FastRam hardwaremäßig abzuschalten, noch einfacher. Allerdings ist eine Erweiterung auf 1,5 MByte nicht mehr so leicht durchführbar wie zuvor.

Eine weitere Änderung: Der Anschluß des Lightpen wurde auf den zweiten Maus-Port gelegt. Mit Hilfe eines Jumpers läßt sich diese Einstellung ändern. Die Entwickler haben auch den Video-Teil durch einen 36poligen Slot erweitert.

Die wichtigen Jumper im einzelnen:

— J101: Mit diesem wird das höchste Adreßbit der Speichererweiterung eingestellt. Momentan ist es A23. Es wird benötigt, um den Speicherbereich ab $C00000 zu verwalten. In der zweiten Position ist es Bit A19. Es erlaubt die Nachrüstung mit einem erweiterten VideoChip. Dieser wird also 1 MByte verwalten können.

— J200: Der Jumper regelt den Lightpen-Port.

— J300: Er ist zuständig für den Takt des CIA Timer Chip. In der momentanen Position wird der Timer mit Netzfrequenz, das heißt 50 Hertz, getaktet. In der zweiten Position bestimmt der Takt des vertikalen Synchronisierimpulses das Geschehen.

— J301: Dieser Jumper ist für den Anschluß eines zweiten internen Laufwerks entscheidend.

— J500: Sind die oberen 512 KByte des 1 MByte großen RAM-Bereichs unerwünscht, können Sie diesen Jumper auftrennen. Dann besitzt der Amiga 2000 nur noch 512 KByte Chip-RAM.

Quelle: Technical Reference Manual 500/2000

(Jörg Koch/ub)

Beschleunigter A/C-Compiler

Der A/C-Basic-Compiler übersetzt Amiga-Basicprogramme in schnellen Maschinencode. Doch auch der Compiler selbst läßt sich beschleunigen.

```
MAKEDIR ram:
COPY source to ram:
CD ram:
df0:AC-Basic
```

Durch diese einfache Befehlsfolge erreichen Sie eine erhebliche Steigerung der Compiliergeschwindigkeit. (Tilo Renkl/ub)

Amiga Speichererweiterung intern, Fastram
1 MB **749.–** 2 MB **1248.–** 4 MB **2048.–**
(Test A-Special 3/87, A-Magazin 12/87)

Laufwerk 1036A, abschaltbar, anschlußfertig
im amigafarbenen Blechgehäuse **329.–**
(Test Kickstart 10/87, A-Magazin 1/88)

Laufwerk 5¼ Zoll, abschaltbar, MS-DOS,
40/80 Tracks, durchgeschleift, TEAC **429.–**
CHINON-Laufwerk, 5¼ Zoll,
genau wie TEAC, beige Frontblende nur **369.–**
Amiga 500, 512 K abschaltbar **229.–**

Amiga 500, 2 MB abschaltbar **919.–**
(Test A-Special 5/87, A-Magazin 12/87)
Amiga Digitizer A500, 1000, 2000 **99.–**

Amiga Pal-Genlock-Interface **549.–**
(Test A-Special 5/87, A-Magazin 2/88)
Coll-Card und Coll-Prom, die ersten Eprom-
karten und Eprombrenner für den Amiga
Coll-Card A500+A1000, 512 K best. **298.–**
weitere 512-K-Karten **49.–** 2 MB **419.–**
Coll-Card A2000 2 MB **398.–**
Coll-Prom A500+A1000 **398.–**
Coll-Prom A2000 **650.–**
Anfragen ab 17.00 Uhr unter 06163/1278 (4326)

FUTUREVISION
Friedrich-Veith-Str. 21, 6128 Höchst/ODW.

AMIGA-SOFTWARE

FASTLIGHTNING

Der neue Maßstab kopiert von DFO auf drei externe Laufwerke in 66 Sekunden 880 KB (gleich eine Kopie in 22 Sekunden), Fastlightning kopiert in 4 Modi bis Track 81 89,– DM

WITHE LIGHTNING mit Formatiermodus und Verify, sehr schnell und sicher. Das Meistbenutzte (Amiga Spez. 4/87) für nur ein Laufwerk 49,– DM

BOOTBLOCK GENERATOR zum Erstellen eines Vorspanns mit Sound. Die erzielbaren Effekte sind einfach gigantisch (Amiga Spez.Nr. 4) 49,– DM

VESALIA VERSAND
G. Does, Marienweg 40, 4230 Wesel,
Telefon 0281/65466 u. 62205



Ware
P. Engels
Postfach 1331
5308 Rheinbach
Tel.: 02226/5714

Amiga-Zubehör von Spezialisten:

A 500 Via-Karte, 40 IOs, 4 Timer etc.,
voll gepuffert: 35/98,– DM

A 2000 RAM-Schalter, schaltet Fast-RAM,
steckbar, fertig: 25,– DM

A 500/2000 Genlock-Interface: 449,– DM

A 500 512-K-Erweiterung schaltbar,
mit Uhr + Accu 35/198,– DM

A 500 Expandermodul, 3 Erweiterungen
am Exp.-Port: 85,– DM

A 2000 PC-Karte inkl. MS-DOS: 1148,– DM

Printer-Switch, 2 Rechner – 1 Drucker
autom. Umschalter: 22/89,– DM

RAM-Chips 41256/150 ns: 6,20 DM

Preise: Leerplatine/Fertiggerät
Durchführung von A 500-Reparaturen!!
Vertriebspartner & Programmierer gesucht!

PUBLIC DOMAIN AMIGA !
3,- 3,- 3,- 3,- 3,- !

Jede 3.5" 2DD/135 tpi Disk z.B. von:
Fish, Panorama, Auge, Faug, ACS, SD,
Amicus, C.Concept., BCS, A. Juice...
Preise je Stück bei Abnahme von :
1-5 4,- 6-10 3,70DM 11 -19 3,50 DM
ab 20 nur noch 3,- ! ab 50 2,90 DM
Gesamtverzeichnis (PDV) auf 3.5" 5,-
(Bar, Briefmarken, keine Schecks !!)
NEU ! PD jetzt auch auf Markendisks:
Fuji, Sentinel, SKC, ö.ä. Aufpr. 1,-

3.5" No Name Disks
2DD 2,19 DM - 2,79 DM

Staffelpreise tel. erfr. Disks sofort
lieferb. Vers.:UPS-Nachn. ca. 8,- VK

Datentechnik M. Bittendorf, Postf.
100248 6360 Friedberg. Tel. von
9-19 Uhr: 06031-61950

Rainbow Data

3½"-Amiga-Laufwerk extern
formschönes Metallgehäuse, helle Frontblende, 880 KB, durchgeführter Port mit Schraubverriegelungen, abschaltbar **329,–**

3½"-Amiga-Laufwerk intern
mit Einbausatz und Anleitung **239,–**

5¼"-Amiga-Zusatzlaufwerk
formschönes Metallgehäuse, helle Frontblende, 40/80-Spurumschaltung, durchgeführter Port mit Schraubverriegelungen, abschaltbar **379,–**

Rainbow Data
Am Kalkofen 1, 5603 Wülfrath
Telefon 02058/1366

INTERNATIONAL SOFTWARE KOELN



Inh.: Elke Heidmüller

AMIGA		AMIGA	
AUTODUELL	64,00	KING OF CHICAGO	59,90
BAD CAT	49,00	LEVIATHAN	57,90
BACKLASH	54,90	MISSION ELEVATOR	49,00
CITY DEFENCE	27,00	PLUTOS	37,90
CALIFORNIA GAMES dt.	64,90	PINBALL WIZARD	45,90
DARK CASTLE	64,90	STRIKE FORCE HARRIER	63,90
D. HAUCH DES TODES 007	49,00	STREET GANG	49,90
EMERALD MINE	27,00	STAR WARS	59,90
FIRE POWER	64,90	SPACE BALLER	28,90
GARRISON	59,00	SPACEPORT	49,90
GOKART RACING	27,00	SPACE RANGER	28,90
GOLDRUNNER	59,00	SUPER HUEY	49,00
INSTANITY FIGHT	64,90	THAI BOXING	28,90
IMPACT	37,90	THE BIG DEAL	66,90
INTERNATIONAL KARATE	58,90	JAGD A. ROTEN OKT. (dt.)	64,90
INDOOR SPORTS	59,90	TERRORPODS	59,90
		WESTERN GAMES	49,90

WEITERE TOP-TITEL AUF ANFRAGE.
ALLE LIEFERBAREN NEUERSCHEINUNGEN VORRÄTIG!
Versand per NN plus 6,50 DM * Preisänderungen vorbehalten

*** 24 Std. Bestellannahme (Anrufbeantworter)**

Computer Softwarevertrieb
Heidenrichstr. 10, 5000 Köln 80
Mo.–Fr. 14.00 bis 19.00 Uhr, Tel. 0221/604493

Bekanntmachung!!
X-tension Harddisks
Made in Germany
20 MB, 40 MB, 72 MB
zu Toppreisen!

Superschnell
durch Interleave 0
Schaltnetzteil u.v.m.

Fa. T. Eder
Peyerstraße 35
8500 Nürnberg 80
0911/329558 17–20 Uhr

AMIGA-SOFTWARE ZU SUPERPREISEN
incl. 2 DD Diskette ab 3.65 DM

Fred-Fish ....**Nr. 1-118** FAUG ....**Nr. 1-39**
PANORAMA ...**Nr. 1- 55** AUGE 4000.......**Nr. 1-12**
Amicus............**Nr. 1- 20** und andere ....
(Katalogdisk gegen 5.- DM anfordern)
Einzeldisk..**4.70 je Disk** ab 40 St.....**4.20 je Disk**
ab 10 St......**4.50 je Disk** ab 100 St....**3.90 je Disk**
ab 20 St......**4.40 je Disk** alle 274 St. zu.....**999.00**
alle Preise incl. 2 DD Diskette

TORNADOS - Die Super PD-Serie. Alle Prg. laufen garantiert auf AMIGA 500/1000/2000.
Nr. 1-30 (incl. 2 DD Disk) 6.00 je Disk
(Katalogdisk gegen 5.- DM anfordern)

BOOTWRITER V 1.0 - kostenlose Infos anfordern **17.- DM**

IFF-CON 0.9 - kostenlose Infos anfordern **27.- DM**

Inland: Porto + Verpackung 3.- DM je Bestellung
Ausland: Porto + Verpackung 6.- DM je Bestellung
(nicht bei Anforderung v. Infos od. Katalogdisk)
Lieferung gegen Vorkasse oder V-Scheck
Achtung! Kein Ladenverkauf!

Bestellung und Anfragen an:
PD-Shop
Opladener Straße 30, D-4018 Langenfeld

NEU! **Ihre AMIGA Grafiken** **NEU!**
auf
DIA/NEGATIV/PHOTO/ OVERHEADFOLIE

INFO: LOFT POST anfordern!!!
tel.: 0561 - 87 33 99

NEU! **NEU!**

STUDIO MAGIC DEMO
+ LOFT POST 1/88
DM 5,-
im Brief (Schein od. Marken) od. KtNr. 61655 Stadtsp.kasse Kassel, BLZ 52050151 od. NN + DM 6,- NN geb.

video LOFT
Fiedlerstr. 22 - 32
D - 3500 Kassel

Mo - Fr 10 - 18.30 Uhr
Sa 10 - 14 Uhr
l.Sa 10 - 18 Uhr

video LOFT film
HARD & SOFTware GmbH

3.5" No Name Disks
2DD 2,19 DM - 2,79 DM

Jede Diskette 100% (!!)geprüft, inkl. farbiger Aufkleber, 10/50er Karton !!
Bitte Staffel-, Tagespreise tel. erfr.

50 2DD ca. 120,-- 100 2DD ca. 230,--
100 5.25" 2D Disks 48/96 tpi nur 79,-
Colordisketten 3.5" schaffen Ordnung:
Rot-Gelb-Grün-Orange-Pink-Weiß......
10 2DD viele v. Farben mgl. 29,90 DM
Diskettenbox für 150 3.5" Disks 45,--

3.5" No Name Disks
2DD 2,19 DM - 2,79 DM

Disketten selbstverständlich sofort ab Lager lieferbar per UPS-Nachnahme

Ausf. Infomat: frank. Rückumschlag
Datentechnik M. Bittendorf, Postf. 100248 6360 Friedberg. Tel. von 9-19 Uhr: 06031-61950

Musik- und Grafiksoftware Shop
Das Spezialgeschäft für Grafiksoft- und Hardware
Wasserburger Landstr. 244 ★ 8000 München 82
Telefon 089/4306207

SCANNER ATARI ST und AMIGA IBM (DIN A4, 200 Dots/Inch)
Flachbett-Scanner, 10 Sek. Scannzeit, mit eingebautem Thermodrucker. Verwendung als Scanner, Kopierer und Hochgeschwindigkeitsdrucker. Mit integriertem Zeichenprogramm für hochauflösende Bilder. Kompatibel zu Degas, MonoStar, Campus, Fleetstreet Publisher, Publishing Partner und vielen anderen Zeichen- und DTP-Programmen, Schriftenerkennung und Telefax-Software in Vorbereitung **2998,–**

Sound-Digitizer für ATARI ST und AMIGA
Digitalisier-Tableau ATARI ST und IBM
Videodigitizer (ATARI/AMIGA/IBM PC AT/C64)
Zahlreiche Grafik- und DTP-Programme für Atari ST und Amiga auf Lager

DIGI-PAINT Das neue 4096-Farben Malprogramm für AMIGA
DELUXE-PAINT II PAL-Version mit deutschem Manual

Fordern Sie unseren kostenlosen Katalog an. Ausführliche INFO zu allen Produkten. Täglicher Versand per Nachnahme oder Vorkasse!

Rufen Sie uns einfach an oder besuchen Sie uns in unserem Laden!
MO-FR 10—18.30 UHR SA 9—13.00 UHR

AMIGA:

Die Massenspeicher kommen...

20-MB-Festplatte, anschlußfertig .... **DM 1398,–**

40-MB-Festplatte, anschlußfertig .... ab **DM 2498,–**

70-MB-Festplatte, anschlußfertig .... **DM 4198,–**

Weitere Größen auf Anfrage!

P M D 08106/33941
Matthias Aures · Postfach 100105
8011 Vaterstetten

AMIGA UTILITIES

GENLOCK 8700 für alle AMIGA	DM 1.095,-
Polaroid Palette m. Interf. Imprint	DM 6.750,-
64 EMULATOR mit Interface	DM 149,-
PERFECT SOUND Stereo Digitizer	DM 225,-
STUDIO MAGIC Digital Sound Studio	DM 129,-
SCULPT 3D m. dt. Handbuch	DM 229,-
SILVER m. dt. Handbuch	DM 349,-
deutsche Handbücher f. Video-scape 3D, Sculpt 3D, Silver: je	DM 39,95
Aztec C Manual in deutsch v. 3.4	DM 128,-

weitere Info: LOFT POST anfordern!!!
tel.: 0561 - 87 79 28 - 87 33 99

video LOFT
Fiedlerstr. 22-32
D-3500 Kassel



Mo-Fr 10-18.30 Uhr
Sa 10-14 Uhr
l. Sa 10-18 Uhr

★P★E★G★G★Y★

Bevor Sie sich irgendwelche Programme wie:

- **Copy-Tool**
- **Disk-Monitor**
- **File-Monitor**
- **System-Monitor**
- **CLI-Tool, Shell**
- **Tasten-Editor**
- **Drucker-Spooler**
- **Task-Handler**
- **Fullscreen-Editor**

oder sonst ein Werkzeug zur Programm-Entwicklung anschaffen, sollten Sie sich, **in Ihrem eigenen Interesse,** über *PEGGY* informieren.

PEGGY, die komfortable Programmier-Umgebung von der ersten Zeile bis zum fertigen Programm. Für alle Programmiersprachen geeignet!

PEGGY 2.12 **nur DM 128,–**
PEGGY's Editor incl. Demo nur **DM 10,–**

SAS-BERND
SYSTEM- & ANWENDERSOFTWARE
Langgasse 93 · 5216 Niederkassel 5
☎ (02281) 452626



ALCOMP
COMPUTERHARDWARE

AMIGA: Speichererweiterung, abschaltbar für 512K zusätzliches RAM (Echtzeituhr nachrüstbar), Komplett mit 512K: **189.-**
Leerplatine + Stecker für AMIGA 500: **39.-**
Uhrenchip 6242 **24.-**
3,5" Laufwerke für Amiga/Atari ST
für Amiga anschlußfertig m. Gehäuse **298.-**
für Atari St w.o. und Netzteil **329.-**
TEAC FD135FN 1MB 2,7cm hoch **239.-**
Soundsampler für alle AMIGA's **79.-**
BESTELLUNG/VERSAND
ALCOMP · A. Lanfermann
Lessingstr. 46 · 5012 Bedburg · Tel. 02272/1580

Rainbow Data

Speichererweiterung für

Amiga 500
512-KB-RAM-Speicherkapazität,
Abschaltung optional **239,–**

Druckerkabel
Amiga 500/2000 **23,–**

Druckerkabel
Amiga 1000 **23,–**

Monitorkabel
Amiga/Scart **29,–**

Rainbow Data
Am Kalkofen 1, 5603 Wülfrath
Telefon 02058/1366

AMIGA-LAUFWERKE

5,25"-Laufwerk MS-DOS und Amiga-DOS-kompatibel, Metallgehäuse und Frontblende in Amigafarbe, abschaltbar mit durchgeführtem Bus 398,- DM

3,5"-Slimline-Laufwerk
Metallgehäuse und Frontblende in Amigafarbe, abschaltbar mit durchgeführtem Bus 329,- DM

3,5"-Laufwerk intern für A 2000 bereits **modifiziert** mit Einbauanleitung 245,- DM

VESALIA VERSAND
G. Does, Marienweg 40, 4230 Wesel,
Telefon 0281/65466 u. 62205

AMIGA 500	1148,-
AMIGA 2000	2398,-
AMIGA 2000 mit 1084	3048,-
AMIGA Colormonitor 1084	698,-
Profex CM-14S Color Stereo! - Mon. inkl. Kabel f. Amiga	648,-
NEC Multisync Monitor	1450,-
Kabel Amiga-NEC Multis.	39,-
VC-1900 Grün-Monitor	189,-
AMIGA 1010 2. Floppy 3.5	469,-
2000 2. Floppy intern	379,-
2000 PC/AT-Karte	1898,-
Commodore MPS-1200	529,-
Epson LX-800	598,-
Star NL-10 inkl. Interface	598,-
NEC-P2200	1098,-
NEC-P6 bzw. MPS-2000	1198,-
NEC-P6 color/MPS-2000C	1495,-
Amiga Software PC-Emulator für Amiga 1000	99,-
Superbase für Amiga	249,-
Logistix für Amiga	359,-
Deluxe Paint II	239,-
DATAMAT/TEXTOMAT Amiga	99,-
ATARI 1040STF mit Monitor	1498,-
Atari ST - NEC Mult.sync. - Kabel umschaltbar alle drei Auflösungen!	79,-
Vortex Drivecard 20 MB f. PCs + Amiga 2000 (PC-Seite)	998,-

Electronic-Bauteile/Data-Becker/Markt & Technik usw.
Bitte fragen Sie an!

Telefon 07541/73122
Versand und Ladenverkauf
REICH-ELECTRONIC
Allinger Str. 86/1, 7990 Friedrichshafen



Hier könnte Ihre Anzeige stehen!

Ihre Ansprechpartner für Minis:

Alicia Clees
089/4613-313

Christine Pfäffinger
089/4613-781

AMIGA

Guru  Meditation

Wer arbeitet, macht Fehler, so auch unsere Redaktion. Leider bleiben dabei auch nicht die Listings verschont, die zum korrekten Ablauf aber fehlerfrei sein müssen. Allerdings haben wir jetzt einen Weg gefunden bei der Übertragung der Programme die Sicherheit erheblich zu steigern. In Zukunft fällt also diese Rubrik (hoffentlich) deutlich kleiner aus.

Biorhythmus mit Extras, Ausgabe 8-9/87, Seite 74:
Lang, lang ist es her, aber der Fehler ist erst jetzt richtig ans Tageslicht gekommen. Das Programm berechnet den Wochentag im Januar und Februar falsch. Die Zeilen 367 und 368 sind durch folgende zu ersetzen:

```
IF m>2 THEN
  m1=m:j1=j
ELSE
  m1=m+12:j1=j-1
END IF
IF mg>2 THEN
  m2=mg:j2=jg
ELSE
  m2=mg+12:j2=jg-1
END IF
a=INT(365.25*j1)-INT(j1/100)+INT(j1/400)+INT
(30.6001*(m1+1)+t
b=INT(365.25*j2)-INT(j2/100)+INT(j2/400)+INT
(30.6001*(m2+1)+tg
```

Außerdem müssen Sie noch die Zeile 374 abändern:

```
n=ABS(n):n=n-7*INT(n/7)+2
```

Nach diesen Änderungen läuft das Programm auch für diejenigen, die im Januar oder Februar geboren wurden.

Raytracer, Ausgabe 11/87, Seite 64:
Hier sollte man die Zeile 16 folgendermaßen abändern:

```
IF b$="n" THEN GOSUB ernst: GOTO Weiter: ELSE GOSUB
test
```

Das Label »Weiter:« muß vor der folgenden Zeile stehen. Manche unserer Leser hatten Probleme, da sie vor dem Programmlauf nicht das Subdirectory »Bilder« angelegt haben. Sie können dies beseitigen, indem Sie in den Zeilen 200 und 213 den Teil

```
"Bilder/"+
```

weglassen. Danach werden die Bilder im Hauptverzeichnis abgelegt.

Mehr Sound im Monitor, Ausgabe 11/87, Seite 102:
Bei der Bauanleitung sind die Werte der Widerstände R1 und R2 leider falsch angegeben. Es handelt sich hier nicht um kΩ, sondern um Ω, also 4,7 Ω und 3,3 Ω. Nach dem Austausch läuft die Schaltung dann richtig.

DCopy, Ausgabe 12/87, Seite 42:
Hier wurde der Befehl zum Linken falsch angegeben. Natürlich lautet der Aufruf

```
ln dcopy.o -lm32 -lc32
```

Checkie42, Ausgabe 12/87, Seite 80 und 1/88, Seite 68:
Gerade bei dem so wichtigen Checksummer hat uns der Guru zweimal denselben Streich gespielt. In Zeile 153 hat er einen Buchstaben zuviel gebracht. Richtig muß es heißen:

```
ELSEIF e=30 THEN
```

Zum Ausgleich hat er in Zeile 162 ein »s« geschluckt. Zum richtigen Ablauf muß hier stehen:

```
GOTO noCrs
```

Sternenhimmel, Ausgabe 12/87, Seite 86:
Dieses Listing ist unser Sorgenkind. Bei der Übertragung passierten mehrere Fehler. Aber die Fehlerquelle wurde jetzt ausgeschaltet. Hier die zu verändernden Zeilen:

```
115 cU0 IF wz<100 THEN wz=wz+2400
138 480 wz$=MID$(STR$(INT((wz-INT(wz))*100+.5)),2)
214 U3  bs=1:frei=1:MOUSE OFF:TIMER OFF:WINDOW 3,,
        (215,70)-(415,112),0,1:WINDOW OUTPUT 3
314 NV  INPUT" ",n$:WINDOW CLOSE 3
530 bO  MENU 1,5,1,"f"+CHR$(252)+"r "+h1$+":"+m1$+"
        zeichnen"
622 HP  DATA 252.2,-34.3,sco5,7
636 bM  DATA 195.5,11.0,vir3,9
691 DK  DATA 231.2,59.0,dra3,19
736 9a  DATA 11.00,.0934,49.326,1.850,1.5237
```

In Zeile 257 ist die nächste Zeile (258) auch enthalten, bitte geben Sie alles nach »258 u5« in einer eigenen Zeile ein.

Amiga-Shell, Ausgabe 12/87, Seite 98:
In Zeile 176 wurde eine Klammer zuviel gedruckt. Richtig lautet die Zeile deshalb:

```
176 zn2 FreeMem(fkey[fk],strlen(fkey[fk]+1);
```

Das im Text beschriebene kopfstehende Ausrufezeichen in Zeile 215, konnte unsere Satzmaschine nicht darstellen. Zwischen die beiden einfachen Anführungszeichen gehört also kein Leerzeichen, sondern dieses Amiga-spezifische Zeichen, das Sie mit <Alt>-<i> erreichen.
Auf der nächsten Seite sind noch zwei Zeilen falsch:

```
386 HV   if (his<0 ) his=hismax;
404 JX   xm=0;x=0;
```

Außerdem müssen Sie beim Linken des Programms den folgenden Befehl verwenden:

```
ln amigashell.o -ls32 -lc32
```

Damit wird die Bibliothek mit den Bildschirmsteuerfunktionen eingebunden.

Joystickabfrage, Ausgabe 12/87, Seite 106:
Hier wurden bei den Aufrufen für die zwei C-Compiler einige Buchstaben ersetzt. Die »1« in den Aufrufen ist immer mit einem »l« auszutauschen.

Tips & Tricks, Ausgabe 1/88, Seite 84:
Im Tip »Grüße aus dem Herzen des Amiga« ging das PEEK verloren. Die vorletzte Zeile lautet korrekt:

```
PRINT CHR$(PEEK(I));
```

Kampf mit harter Hand, Ausgabe 11/87, Seite 23:
Man kann Karate Kid II natürlich auch zu zweit spielen. Dazu drückt man zum Starten des Spiels den Knopf des Joysticks, der an Port 1 (normalerweise der Mausanschluß) angeschlossen ist. Der zweite Joystick steckt wie immer in Port 2.

Die Public Domain-Seite

Mit Hilfe von Freesoft-Disketten können sehr preiswert nützliche Programme erworben werden. Aus über 350 Disketten läßt sich hierbei schöpfen. Wir sagen Ihnen, was für Sie interessant sein könnte.

Fred Fish läßt nicht locker: Kurz vor Redaktionsschluß konnten wir erfahren, daß die Fish-Serie inzwischen auf 118 Disketten angewachsen ist. Auch andere Serien haben Zuwachs erhalten. So etwa die ACS-Serie, die nun 15 Disketten umfaßt.

Die »ACS 11« bis »ACS 13« sind Zusatzdisketten für IFF-Musikprogramme wie etwa Sonix. Auf diesen drei Disketten finden sich viele neue Instrumente und eine Menge Melodien, die teilweise sehr gut klingen. Wie auch schon mit der »ACS 10« findet sicher jeder musikinteressierte Anwender großen Gefallen an diesen drei Disketten, die wir jedem Musiker empfehlen können.

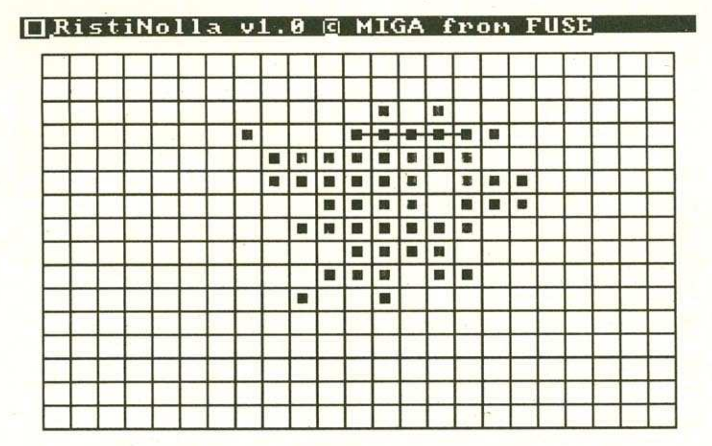
Die weiteren neuen ACS-Disketten mit den Nummern 14 und 15 sind eher für Grafiker interessant. Viele, meist neue IFF-Bilder sind dort in einer Diashow zusammengefaßt. Die Bilder bieten zwar nichts Außergewöhnliches, sind aber in einer IFF-Bildersammlung nicht fehl am Platz.

Eine andere Serie ist die des »Sacramento Amiga Computer Clubs« (SACC). Die »SACC 2« enthält eine gute Auswahl an verschiedenen Programmen. Da wäre zum einen eine IFF-Bildersammlung mit sehr guten, teilweise digitalisierten Bildern und einem entsprechenden Slideshow-Programm.

Viel Software von A bis Z

Des weiteren findet sich die resetfeste ASDG-RAM-Floppy, eine RAM-Disk, die auch bei einem Reset die Daten nicht verliert. Interessant ist auch die kleine Adressenverwaltung »BlackBook«, mit der leicht persönliche Anschriften gespeichert werden können. Das bekannte Spiel »CosmoRoids« ist ebenso enthalten wie ein gutes Biorhythmusprogramm. Auch für diese Diskette gilt: Nicht lebensnotwendig, nimmt aber auch in der PD-Sammlung keinen Platz weg.

Sehen Sie sich gerne Grafiken an? Die »Digital Grafik-Disk 1« von 22-Software enthält viele digitalisierte Grafiken im Normal- wie auch im H.A.M.-Modus. Seien es Landschaftsansichten oder Aufnahmen von Gebäuden. Die Bilder bestechen zwar nicht durch Schärfe, können aber doch in eine Grafiksammlung aufgenommen werden.



Ein Spiel, das Freude bringt: »RistiNolla«. In kleinerer Ausführung (mit 3 x 3 Feldern) auch als Tic Tac Toe bekannt.

Viel Neues findet sich auf Fish »103«. Beispielsweise das Programm »IntuiDOS«, das eine Art dritter Benutzeroberfläche darstellt. Über dieses Programm können unter anderem durch einfaches Anklicken Dateien ausgeführt (EXECUTE), Files und Directories angezeigt (LIST) oder diverse andere Kommandos gegeben werden.

Weiter enthalten ist ein Packer, mit dessen Hilfe bis zu 2 MByte auf eine Diskette gespeichert werden können. Auch ein Update der Demo-Version des Microfiche-Filers (eine kleine Datenbank) fehlt nicht. In dieser Datei enthalten ist ein Katalog der meisten PD-Disketten mit kurzen Kommentaren zu den einzelnen Programmen. C-Programmierer werden sich dagegen sicher für »CRef« interessieren, einem Programm, mit dem sich für C-Sources Cross-Referenz-Listen erstellen lassen. Diese Diskette ist durch die Vielfältigkeit der enthaltenen Programme bestens zum Aufbau einer PD-Sammlung geeignet.

Sicher haben Sie sich schon gelegentlich Gedanken darüber gemacht, daß es manchmal sinnvoll wäre, den Amiga oder spezielle Disketten gegen Benutzung von anderen zu schützen. Fish »105« enthält dafür das Programm »FlamKey«, welches die Maus und die Tastatur über ein Paßwort vor fremder Benutzung schützt. Auch das Gag-Programm »DrunkenMouse«, das einen torkelnden Mauszeiger erzeugt, ist hier zu finden. Programmierer haben sicher auch ihre helle Freude an dieser Diskette. Verschiedene Assembler- und Basic-Programme mit kommentierten Quelldateien erleichtern den Einstieg in diese Programmiersprachen, denn schon durch Nachvollziehen fremder Programmierstrukturen läßt sich sein eigenes Programmiergeschick verbessern. Die Diskette wird durch das Spielprogramm »Gravity Wars« angenehm abgerundet, so daß die Fish »105« keinen Platz in einer PD-Sammlung wegnimmt.

Spielen Sie gerne? Dann gefällt Ihnen sicher »RistiNolla« auf Fish »106« (siehe Bild). Bei diesem altbekannten Spiel geht es darum, auf einem 16 x 23 Felder großen Spielbrett fünf Steine so zu plazieren, daß sie in einer Reihe liegen. Dabei ist es egal, ob diagonal oder gerade. Aber Achtung: der Computer spielt nicht schlecht! Außerdem enthalten: verschiedene IFF-Bilder, ein Update des Programms »Func-Key«, mit dem sich bis zu 50 Funktionstasten belegen lassen und verschiedene Animationsdemos, mit dem sicher der eine oder andere beeindruckt werden kann.

Grafikfreaks werden die Fish »109« mögen. Dort findet sich ein mit Videoscape 3D erstelltes Animationsdemo, auf dem sich sehr viel bewegt. Zum »Überleben« braucht man diese Diskette sicher nicht, aber sehenswert ist das Animations-Demo auf alle Fälle.

Preiswerter Assembler

Ebenfalls für Programmierer ist die Fish »110« hochinteressant. Man findet auf dieser Diskette einen 68000-Assembler und einen optimierenden C-Compiler. Da uns die Disketten erst zum Redaktionsschluß erreichten, konnten wir noch nicht die Leistungsfähigkeit dieser beiden Programme erforschen. Wir werden uns aber näher mit diesem Programm befassen, um Ihnen in Ausgabe 3/88 des AMIGA-Magazins darüber zu berichten, ob der Compiler oder der Assembler eine brauchbare Alternative zu teuren, kommerziellen Produkten sein kann. Auf jeden Fall aber ist diese Freesoft-Diskette sicher kein Fehlkauf, da sich verschiedene Programme damit erstellen lassen. (dm)

Anbieter von Public Domain-Disketten (ohne Anspruch auf Vollständigkeit):
Stefan Ossowski, Veronikastr. 33, 4300 Essen, Tel. 0201/788778
Atlantis, Ernst-Reuter-Str. 151, 5030 Hürth 8, Tel. 02233/31066
Datentechnik Bittendorf, Postfach 100248, 6360 Friedberg, Tel. 06031/61950
Fastworks, Fichtestr. 16, 5090 Leverkusen 1, Tel. 0214/92802
EcoSoft AG, Kaiserstr. 21, 7890 Waldshut, Tel. 07751/7920
Christian Bellingrath, Trift 10, 5860 Iserlohn, Tel. 02371/24192
Intersoft, Nohlstr. 76, 4200 Oberhausen 1, Tel. 0208/809014-809015
Rainer Wolf Soft- und Hardwareversand, Deipe Stegge 187, 4420 Coesfeld, Tel. 02541/2874
Uwe Schmielewski, Haroldstr. 71, 4100 Duisburg 1, Tel. 0203/376448
Soyka Datentechnik, Hattinger Str. 685, 4630 Bochum 5, Tel. 0234/411913
AB Computersysteme, A. Büdenbender, Wildenburgstr. 21, 5000 Kölm 41, Tel. 0221/4301442
C.A.S., Sprendlinger Landstr. 71, 6050 Offenbach, Tel. 069/842013
22-Software, Thomas Wirz, Höhenweg 98, 5300 Bonn 1

Der 1901 am Amiga

Und es geht doch! Der Commodore-Monitor 1901, bekannt vom C 64 und C 128, läßt sich doch an den Amiga anschließen. Aber damit nicht genug, er ist sogar in der Lage, alle 4096 Farben darzustellen. Und all das in derselben Qualität wie der normale Amiga-Monitor 1081. Für Besitzer eines 1901 eine billige Alternative.

Normalerweise hat der Monitor 1901 nur einen Anschluß für ein RGB-Digital-Signal mit zusätzlicher Intensität. Das heißt, es lassen sich nur 16 Farben darstellen. Aus diesem Grund haben wir bisher keine Anleitung gebracht, wie der Monitor anzuschließen ist.

Eine Entdeckung im Monitor macht es jedoch möglich, diesen als RGB-Analog-Monitor anzuschließen. Dadurch können (fast) beliebig viele Farben generiert werden, auf alle Fälle die 4096, die der Amiga zur Verfügung stellt.

Die Qualität der Bilddarstellung (siehe Bild) steht der des Monitors 1081, der normalerweise am Amiga angeschlossen ist, in nichts nach. Auch der hochauflösende Modus (640 Punkte horizontal) ist qualitativ einwandfrei. Man kann also ohne Probleme auch die 80-Zeichen-Darstellung lesen. Leider tritt im Interlace-Modus (wie beim 1081 auch) das störende Flimmern des Bildes auf. Dieses Problem haben aber alle Monitore, die nicht langnachleuchtend sind. Insgesamt ist das Bild des 1901 qualitativ eher besser als das des Amiga-Monitors 1081.

Zum Umbau des Monitors müssen Sie Erfahrung bei Hardware-Basteleien mitbringen, denn leider geht es nicht ganz ohne Löten. Außerdem ist im Monitor auch nach längerem Warten immer noch Hochspannung vorhanden. Es ist somit extreme Vorsicht geboten. Zusätzlich müssen Sie wissen, daß die Garantie durch den Eingriff erlischt.

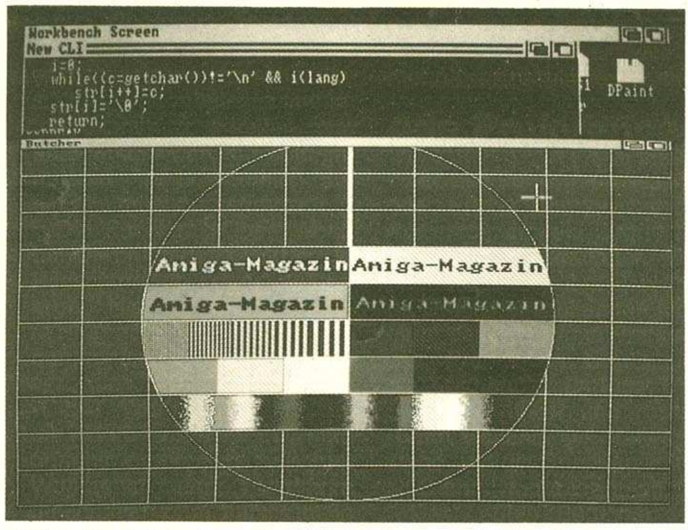


Bild 1. Die Bildschirmdarstellung mit dem Monitor 1901 von Commodore am Amiga-RGB-Anschluß ist einwandfrei

Vorsicht: Lebensgefahr

Schauen Sie sich zunächst die Rückseite des Monitors an. Unten, zwischen den zwei Cinch-Anschlüssen, sehen Sie ein Rechteck, das zum Herausbrechen vorbereitet ist. Dort hinein paßt genau eine SCART-Buchse. Sie können aber die Buchse auch an einem Kabel anbringen, das mindestens 21 Adern besitzen muß.

Nach dem Öffnen des Monitors können Sie die Platine ein Stück herausziehen. Nun stellt man aber leider fest, daß die auf der Platine dahinter vorhandenen Lötpunkte nicht zu einem SCART-Platinenstecker passen. Die zwei Anschlußreihen sind nämlich genau vertauscht. Deshalb müssen Sie unbedingt eine SCART-Buchse kaufen, an die Sie Drähte anlöten können. Machen Sie diese nicht zu kurz, da sonst später das Löten unmöglich wird. In Bild 2 sehen Sie die Belegung der Lötkontakte auf der Platine. Dasselbe Bild müssen Sie sehen, wenn Sie hinter dem Monitor stehen und von oben hineinschauen. Dabei sehen Sie dann die Bestückungsseite der Platine im Innern des Monitors.

Die Belegung der SCART-Buchse von der Lötseite her sehen Sie in Bild 3. Es müssen nun einfach alle Kontakte mit derselben Nummer verbunden werden. Der Anschluß 21 im Bild 3 ist das Abschirmungsgehäuse der SCART-Buchse. Es sind also nur zwanzig Anschlußstifte vorhanden.

Nochmalige Kontrolle

Der Anschluß mit der Nummer 21 muß dabei mit der Abschirmung des Kabels verbunden werden. Dies ist nötig wegen eventueller Fremdeinstrahlungen, die sonst die Bildqualität mindern würden. Nach nochmaliger Überprüfung können Sie dann den Monitor an Ihrem Amiga anschließen. Sie werden dann mit einem guten Bild belohnt, das dem des 1081-Monitors in nichts nachsteht.

(aw/rb)

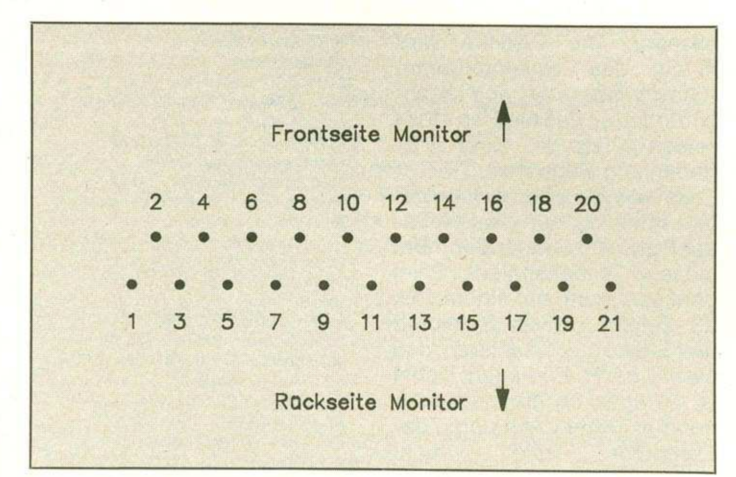


Bild 2. So sehen Sie die Lötpunkte, wenn Sie hinter dem Monitor stehen und von oben auf die Platine blicken

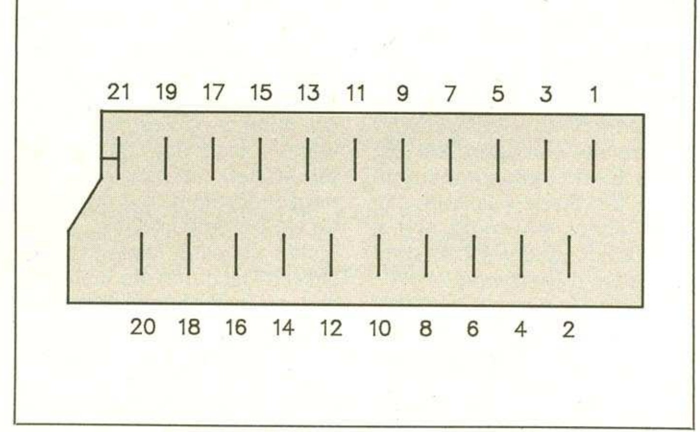


Bild 3. So erscheinen die Anschlüsse einer Standard-SCART-Buchse von der Lötseite her gesehen

Chancen von Lernsoftware

Das Angebot an guten deutschen Lernprogrammen für den Amiga ist praktisch gleich Null. Dennoch laufen Versuche, diesen Mißstand mit wissenschaftlichen Methoden zu beseitigen.

Lernprogramme haben mit dem Erscheinen der Mikrocomputer neue Hoffnungen geweckt. Die Prognosen der Wissenschaftler und Praktiker waren vor noch nicht allzu langer Zeit — zumindest in den USA — sehr positiv. Die Ernüchterung stellte sich in den letzten Jahren ein. Die übersteigerten Erwartungen konnten die Mitte der 80er Jahre im amerikanischen Schulwesen relativ weitverbreiteten Mikrocomputer-Lernprogramme nicht erfüllen. Wie nicht selten bei der hastigen Entwicklung neuer Technologien besteht die Gefahr, durch wenig durchdachte Konzepte bei der Programmentwicklung gute Chancen zu verspielen. Im deutschen Raum ist dies glücklicherweise noch nicht in dem amerikanischen Ausmaß geschehen. Man hat also Zeit, das Ganze mit dem Hintergrund der amerikanischen Erfahrungen neu zu überdenken.

Die Chancen?

Worin liegen eigentlich die Chancen von Lernprogrammen? Diese Frage läßt sich nur mit dem Hintergrund zahlreicher Untersuchungen zur Gestaltung produktiver Lernumwelten und der Theorie der menschlichen Informationsverarbeitung beantworten. Dazu kommt natürlich der pragmatische Aspekt, demzufolge Lernprogramme nur dann einen bedeutsamen Stellenwert im schulischen und auch außerschulischen Unterrichtsgeschehen erlangen können, wenn sie zur Lösung didaktischer Probleme, wie etwa der Verwirklichung des entdeckenden Lernens, beitragen. Dabei sollten sie den Lehrern und Schülern soviel kreative Möglichkeiten wie möglich eröffnen. Das ist besonders wichtig für die Lehrer, denn das Gleichgewicht zwischen einengenden Vorgaben und seinen kreativen Gestaltungsmöglichkeiten darf durch Lernprogramme nicht noch weiter in Richtung auf unkreatives Aufnehmen verschoben werden.

Im Umfeld eines an der Universität Köln seit Ende 86 laufenden Forschungsprojektes (Schüler erfinden Lernspiele — Ein Programm zur Förderung des erfinderischen Problemlösens) wurde die Erfahrung gemacht, daß das Erfinden von Spielen den menschlichen »Erfindergeist« auf eine allgemeine und besonders intensive Art herausfordert. Oftmals produzierten Schüler spontan mindestens ebenso viele Ideen zur Verbesserung der entwickelten Spiele wie zu den Spielen beziehungsweise Spielstrategien selbst.

Die Beobachtung, durch Spiele in einer besonders intensiven Weise zunächst die Verbesserungsmotivation zu aktivieren um daraufhin den Erfindergeist zu mobilisieren, wird durch Alltagserfahrungen weiter gestützt. Es gibt wohl in keinem anderen Bereich so viele Laienerfinder wie im Spielebereich. Die Entwickler neuer Spiele sind durchaus nicht nur Erzieher oder Spielenthusiasten, denn schon von etwa 4 Jahren ab erfinden Kinder ihre eigenen Spiele und auch das hohe Alter schützt nicht davor, etwas vordergründig so Nutzloses zu erfinden. Die vielen tausend Spielevorschläge, die jährlich bei den Spieleverlagen eingehen, und auch der aktuelle Boom auf dem Computerspielmarkt, belegen dies in eindrucksvoller Weise.

In der Durchführung des oben genannten Projektes sollen in einem »Denkspielstudio« entwickelte Spiele unter dem Aspekt erprobt werden, diese Spiele zu verbessern, um anschließend zu versuchen, neue Spiele zu entwerfen. Das Spektrum der Spiele soll von den Denkspielen auf Lernspiele zu schulischen Inhalten erweitert werden.

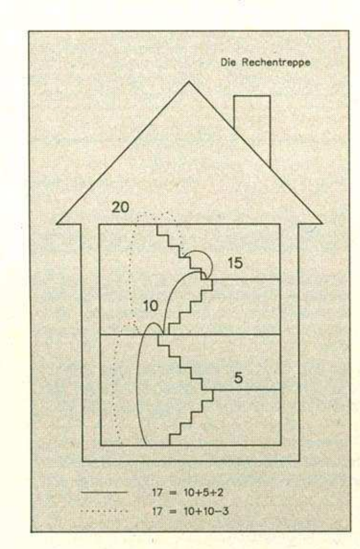


Bild. Die Analogie Rechentreppe

Für die Realisierung der Computerlernspiele ist ein sogenanntes »Autorensystem« entwickelt worden. Es handelt sich um ein vollständig naives System. Es läßt sich ohne jegliche Programmierkenntnisse einsetzen. Ohne jede Computererfahrung lassen sich damit die Ideen zu neuen Lernspieltests oder Lernprogrammen auf dem Computer realisieren. Ein Problem dieses Autorensystems liegt darin, daß nicht alle kreativen Ideen verwirklicht werden können. Für die Schulung des erfinderischen Problemlösens kann das allerdings auch als Vorteil interpretiert werden. Der Erfinder muß seine Ideen auf das mit dem Computer Machbare zuschneiden.

Das Programm

Das Autorensystem trägt den Namen GALA (Gamelike Amiga Learning Adventures). Wie funktioniert es?

Der Autor (Schüler) entwirft mit einem einfachen Zeichenprogramm Symbole, Figuren oder Texte. Ein integriertes Konvertierungsprogramm ermöglicht die Verwendung von IFF-Dateien. Solche Dateien lassen sich mit handelsüblichen Zeichenprogrammen wie »DPaint«, »Prism« oder »DigiPaint« erzeugen.

Die Bildbausteine besitzen Attribute, die festlegen, in welcher Weise der Baustein auf dem Bildschirm agiert oder reagiert. Die Attribute bestimmen, wo sich eine Figur zu einem festgelegten Zeitpunkt befindet, ob sie sichtbar ist oder eine bestimmte Lautfolge sagen soll. Durch Anklicken der Symbole mit der Maus reagiert das Programm entweder durch Beenden des Ablaufs oder Start einer weiteren Folge. Der Autor kann sich Spielfiguren, Spielbrett und Spielregeln selbst entwerfen. So entsteht ein Spielgeschehen nach seinen Vorstellungen.

Das im folgenden vorgestellte Lernprogramm ist eine Anwendung des Lernens durch Analogien. Es hat das Ziel, Verarbeitungsstrukturen und -prozesse bei Kindern mit Hilfe von (Alltags-)Analogien aufzubauen, die zum Rechnen mit den vier Grundrechenarten befähigen. Es wird eine Rechentreppe dargeboten, die wie ein Treppenhaus aussieht (Bild). Sie ist unterteilt in Absätze (nach jeweils fünf Stufen) und Etagen (nach jeweils 10 Stufen), wobei die Stufen genau übereinanderliegen. Darauf agieren die mit Sprungschuhen und Hochsprungstab ausgestatteten Handlungsträger (Figuren) Hase und Igel. Sie können sich auf der Rechentreppe mit Einzelschritten, 5er- und 10er-Sprüngen fortbewegen. Die Kinder erkennen das vertraute Schema »Treppensteigen«. Damit können nach einem spielerischen Vortraining die Handlungsanalogien für die Grundrechenarten eingeübt werden:

- ☐ Addition durch Weiterspringen
- ☐ Subtrahieren durch Herunterspringen
- ☐ Multiplikation und Division durch gleichgroße Sprünge

Mit der Vorgabe einer bestimmten Positionierung (zum Beispiel auf Stufe 17) können nun verschiedene Strategien zur Erreichung dieses Ziels entdeckt werden. Erklärt man die Absätze und Etagen als Ausruhpunkte, so läßt sich die leichtere Rechnung mit »glatten Zahlen« demonstrieren.

Mikrocomputer für Systeme wie GALA müssen eine leistungsfähige Hard- und Software-Unterstützung grafischer Objekte, sowie eine einfache Bedienerführung gewährleisten. Damit ist der Amiga für solche Anwendungen sehr gut geeignet. Die Entwicklung in den nächsten Jahren wird zeigen, mit welchem Erfolg solche Projekte in die Tat umgesetzt werden.

(Dr. H. Rüppell/L. Galke/pa)

Gala-Software, Lioba Galke, Goetheweg 3, 5630 Remscheid 11

AMIGA COMPUTER-MARKT

Wollen Sie einen gebrauchten Computer verkaufen oder erwerben? Suchen Sie Zubehör? Haben Sie Software anzubieten oder suchen Sie Programme oder Verbindungen? Der COMPUTER-MARKT von »Amiga« bietet allen Computerfans die Gelegenheit, für nur 5,— DM eine private Kleinanzeige mit bis zu 4 Zeilen Text in der Rubrik Ihrer Wahl aufzugeben. Und so kommt Ihre private Kleinanzeige in den COMPUTER-MARKT der **April-Ausgabe** (erscheint am 23. März 88): Schicken Sie Ihren Anzeigentext bis zum 17. Februar 88 (Eingangsdatum beim Verlag) an »Amiga«. Später eingehende Aufträge werden in der **Mai-Ausgabe** (erscheint am 27. April 88) veröffentlicht.

Am besten verwenden Sie dazu die vorbereitete Auftragskarte am Anfang des Heftes. Bitte beachten Sie: Ihr Anzeigentext darf maximal 4 Zeilen mit je 40 Buchstaben betragen. Überweisen Sie den Anzeigenpreis von DM 5,— auf das Postscheckkonto Nr. 14199-803 beim Postscheckamt mit dem Vermerk »Markt & Technik, Amiga« oder schicken Sie uns DM 5,— als Scheck oder in Bargeld. Der Verlag behält sich die Veröffentlichung längerer Texte vor. Kleinanzeigen, die entsprechend gekennzeichnet sind, oder deren Text auf eine gewerbliche Tätigkeit schließen läßt, werden in der Rubrik »Gewerbliche Kleinanzeigen« zum Preis von DM 12,— je Zeile Text veröffentlicht.

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen

Suche: Software

Suche AKTUELLE Amiga-Soft. Tausch auch möglich. Auch Public Domain. Habe auch noch zirka 800 Zeitschriften. Günstige Angebote an Heike M., Tannenbaum 17, 418 Goch 2

Wenn der Computer streikt, helfen sich die DEHOCA-User untereinander oder nutzen das bundesweite, dezentrale Servicenetz, Info: Postf. 1430, 3062 Bückeburg

»Suche Software und Tauschpartner für Amiga 1000. Bitte schreibt an: Andreas Zeising, Sonnenleite 4, 8913 Schondorf«

● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●
Suche Tauschpartner:
0451/603378 Mathias
● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●

Suche günstig Software für Amiga 500 (Anwendung, Grafik, Musik, Spiele, PD, etc.) Angebote bitte an P. Norrman, Ackerstr. 199, 4000 Düsseldorf 1

★ ★ ★ AMIGA-SOFTWARE-TAUSCH ★ ★ ★
Suche noch Tauschpartner, Top-Software vorhanden. Antworte schnell! Roberto Mattei,
★ ★ Urbanusstr. 28, 4650 Gelsenkirchen 2 ★

★ ★ Kontakte ★ ★ zu Amiga-Usern gesucht. Software- und Erfahrungsaustausch, Lothar Urbas, Brüder-Grimm-Str. 9, 3584 Zwesten, Tel. 05626/1679 (ab 17 Uhr)

Suche für meine Sammlung noch folgende ORIGINALE (keine Raubkopien!!!): Faery Tale, Uninvited, und speziell Infocoms, Tel: 05204/8196 (nach 20.00 Uhr)

DEHOCA-Service »Public-Pool«. Für alle Mitglieder vermitteln wir die besten Tagespreise auf Hardware — neu oder gebraucht. Info: Postf. 1430, 3062 Bückeburg

■■ AMIGA ■■ MS-DOS ■■ AMIGA ■■
Tausche neueste Amiga- und MS-DOS Software. Habe/Tausche auch Anleitungen!!!
★ Tel. (Germany) 05561/6915 NEW MAN ★

Amiga-Einsteiger sucht Software und Kontakt zu anderen
Andreas Kilgenstein, Böhmerwaldstr. 9
8011 Neukeferloh, Tel. 089/466140

Wichtig: Gibt es schon Turbo-Pascal für den 500er? Wenn ja, wo + wie? 04120/671 ab 15 Uhr. R. Schwerdtfeger, Ahrenloher Str. 181, 2082 Tornesch ★ HELP ★

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★
We are searching for new contacts!
Call: 05732/3455 (DIRK)
Only between 18—21! Eddie is alive!

Suche für Amiga 500 Software zum Tauschen ★ ★ Kaufe keine Programme ★ ★ möglichst im Raum Essen. Listen an: Reiner Osenar, Zweigstr. 42, 4300 Essen 11

T.Sk. sucht neue Kontakte auf Amiga, neueste Software vorhanden, keine finanz. Int., Tel. 05621/73417

Suche günstige Software für Amiga. Besonders Grafiksoftware + Desktop Publ., Listen an Mathias Wendel ★ ★ ★ Bunnenbergstr. 24, 3000 Hannover 1

Amiga-Fan sucht günstige Software (PD). Sendet Eure Listen an Friedrich Neuper, 8473 Pfreimd, Leuchtenberger Str. 1

DRINGEND: Suche f. Amiga 1000 Extras Diskette Basic-Tutor 1, 2 gegen Bezahlung versteht sich. Tel. 0211/7335982 Frl. Otte

Ich suche Software für meinen Amiga 500. Schreibt an: K. Baumberger, Lavendelweg 38, 2000 Hamburg 70
Wer hat Börsenprogramme?

Mailbox-Freaks drucken sich ihre DEHOCA-Beitrittserklärung selbst aus. Zu finden in allen DEHOCA-Regionalboxen und natürlich in der Verbands-Zentralbox 05722/3848

Suche Tauschpartner für Amiga 500 aus aller Welt.
Schreibt an: Michael Urban, Große Straße 27
4530 Ibbenbüren 1

Suche günstige Software für Amiga (bitte mit Anleitungen). Listen an Michael Gschwender, Dorfstr. 3, 8968 Durach

Amiga 500: Suche preisw. Software für Heilpraktiker, z.B. Prüfungsvorbereitung, Homöopathie usw. G. Wincheringer, Sponheimer Str. 55, 5580 Traben-Trarbach, Tel. 06541/9886

Amiga Anfänger sucht, tauscht, kauft Software für Amiga 500. Sofort anrufen bei Uwe 04681/8148!!!

Suche Aktienprogramm für meinen Amiga 1000, wenn möglich mit Grafik, evtl. Logistix oä., zahle/tausche gut. Robert Hermann, Schloßstr. 24, 8851 Marxheim/Sch.

Suche billige Public-Domain-Software für A2000. Immer Antwort! Hauke Arndt, Stadtweide 15, 2400 Lübeck

Amiga 500-Freak sucht Tauschpartner (auch für Public-Domain)
Frank Balmes, Hauptstr. 1
5452 Weißenthurm, Tel.: 02637/2544

A-500 Anfänger sucht Tauschpartner für Software. Bin besonders interessiert an Spielen, Musik- und Grafikanwendungen. René Schätzl, Trift 1A, 6534 Roth

A500-Anfänger sucht Software; Anwender! Bsd. Textverarb., DTP, Dpaint II, Dprint II, sowie Business-Soft jed. Art, zahle gut! M. Bretag, Haidbarg 5, 2110 Buchholz 04181/31189

Amiga 500 User
Für Kauf von Public Domain-Software — suche wegen Kostenteilung Interessenten. Lichter, Braunfelser Str. 13, 6330 Wetzlar

Suche Software für Amiga. Liste bitte an Bormes Günter, Hauptstr. 10, 5523 Rollersdorf

A-2000 Aufsteiger sucht Prg.s und Kontakte zu Amiga-Usern ■ Raum PLZ 2000 !!! Suche insbesondere Lattice C oder Aztec C sowie Superbase!!! Tel.: 04122/8787

Suche Software für Amiga 500 (nur Spiele/Originale!!) Suche besonders Land of the Lounge Lizards!! Angebote an: Can Imre, Konrad-Adenauer-Ring 11, 8592 Wunsiedel

Suche, tausche, kaufe alle mögliche Software. Listen an Uwe Spindler, Poststr. 16, 3522 Bad Karlshafen 2 oder T. 05672/2236 nur am Wochenende (öfter versuchen)

AMIGA-Einsteiger sucht Software aller Art, vor allem Spiele. Listen an: Peter FACH, Schildweg 61, 3200 Hildesheim
Gerät: Amiga 500

● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●
★ Tausche neueste Amiga Software ★
/7518 05103/7518 05103/7518
● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●

■ AMIGA 500 ■ Call Germany ■ AMIGA 500
■■■ 0251 ■■ 82947!! ■■ (Daniel) ■■■
■■■ I'm searchin' 4 AMIGA-USER ■■■
■■■■■ **all over the globe!!!** ■■■■■

Suche Software für Amiga 500. Schickt Liste an: F. Herter
Plötzenseestr. 17
4019 Monheim 1

Suche zuverlässige Tauschpartner für Amiga 500 Softw.; schickt Eure Liste an Frank Herter, Plötzenseestr. 17, 4019 Monheim 1, 100% Antwort

Suche dringend professionelle (!) Chemie-Biologie-Programme auf dem Amiga 1000! Bitte nur Originale und in Deutsch!
Angebote an Stefan Nürnberg: 02238/14281

We are looking for great contacts to exchange experience and software. Call 0991/23693 (sudl) or write to: Hufo, Steinbühl 1, 8351 Bernried 1

Suche für A-500 Anwender-Software, Spiele und sonstige Programme. (Videoverwaltung, Lager, Grafik usw.) Angebote an: Diepold, Am Kapuzinerhölzl 1, 8000 München 50

Suche Aztec-C V3.4 mit Anleitung
verkaufe Monitor 1081 (1 Monat alt) 600 DM, entferne kostenlos SCA/FST-Viren
Tel. 08702/1764 nur von 19—20 Uhr

★ ★ ★ Suche Tauschpartner für die ★ ★ ★ neuesten Programme. Amiga + C64. Ruf: 04131/62742 (Thorsten)

Anfänger sucht Software für Amiga 500, Tel. 02134/30435 (Dirk) ab 18.00

Amiga-Software
Suche und tausche Amiga-Software
Tel. 0203/722197

Ausland

Suche günstige Floppy sowie Drucker. Suche außerdem noch C64.
Tel. 01/7341645 ab 17 Uhr (Schweiz) Th. Kraushaar, Weiherstr. 67, 8902 Urdorf

Suche/tausche neue Software. Auch PD; Weekend (ab Fr. 19 h) call
Darth Vader. Tel. Austria = 0043/022312850. Bis bald!

Suche Software für A-2000
Meine Adresse: Raymar Knupfer,
Seenerstr. 166
Tel. 052/281612 / 8405 Winterthur (CH)

Suche, tausche Software
Raymar Knupfer
Seenerstr. 166
CH-8405 Winterthur (CH) Tel.: 052/281612

Suche Tauschpartner für Amiga 500!
Top Amiga Games vorhanden!!
Listen an Thomas Landl, Vinzentinum
Bärengäßchen 6, A-5020 Salzburg

Amiga ★ ★ Österreich ★ ★ Amiga
Suche allerneueste Amiga-Software aus aller Welt! Martin GLASER, Nußdorferstr. 52/8, A-1090 Wien, Tel. 0222/317415

★ ★ ★ AMIGA DENMARK ★ ★ ★
Suche Tauschpartner in aller Welt. Software oder Liste an: Hans Erik Pedersen, Boerupvej 88, DK 83/0 Tranbjerg J.

Swap Amiga Stuff. Aztec-C-Compiler (Commercial) for sale. Insert 80 Pf. Write in Deutch or English only! Marcel Erkel, Bij de Kerk 1A, 2965 AA Nieuwpoort-Holland

Suche dringend: MS-DOS Emulator + Prog.-Sprache Cobol für A500 (Emulator unter Kick 1.2). Angebote an Stefan Lussi, Finkenweg 3, CH-3252 Worben, Tel. 032/846251

Hey Amigos!
Suche Tauschpartner für Amiga 500
Vladimir Kozák — Belehradská 27 —
PRAHA 2-12000-CSSR

Kauf oder Tausch von Amiga 500-Software
Hanny v.d. Heuỳ, Michiel de Ruyterstr. 5
NL-5831 KE BOXMEER HOLLAND

★ ★ ★ ★ DANISH DRAGONS IS ★ ★ ★ ★
Searching for new Contacts all over the world. Write to: Ole Johansen, Juivej 13, 6400 Sonderburg, Denmark

Biete an: Software

★ ★ ★ Amiga Grafik u. DTP Freaks ★ ★ ★
Umfangreiche Grafik (1000 Bilder) u. Zeichensatz (150) Sammlung. Info gegen Rückporto. R. Hobbold, Gleisstr. 14, 43 Essen 11

AKTUELLE Amiga-Soft zum absoluten Schleuder-Preis. Freue mich über jeden neuen User.
R.P., Brückenstr. 10, 4180 Goch 1.

Achtung:

Wir machen unsere Inserenten darauf aufmerksam, daß das **Angebot**, der **Verkauf** oder die **Verbreitung** von urheberrechtlich geschützter Software nur für Originalprogramme erlaubt ist.

Das Herstellen, Anbieten, Verkaufen und Verbreiten von **»Raubkopien«** verstößt gegen das Urheberrechtsgesetz und kann straf- und zivilrechtlich verfolgt werden. Bei Verstößen muß mit Anwalts- und Gerichtskosten von über DM 1 000,— gerechnet werden.

Originalprogramme sind am Copyright-Hinweis und am Originalaufkleber des Datenträgers (Diskette oder Kassette) zu erkennen und normalerweise originalverpackt. Mit dem Kauf von Raubkopien erwirbt der Käufer auch kein Nutzungsrecht und geht das Risiko einer jederzeitigen Beschlagnahmung ein.

Wir bitten unsere Leser in deren eigenem Interesse, Raubkopien von Original-Software weder anzubieten, zu verkaufen noch zu verbreiten. Erziehungsberechtigte haften für ihre Kinder.

Der Verlag wird in Zukunft keine Anzeigen mehr veröffentlichen, die darauf schließen lassen, daß Raubkopien angeboten werden.

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen

Tausche Originale: Balance of Power, Terrorpods, Leviathan und andere
Tel. 0203/781649 (Stefan)

■■■■■■■ Amiga 500 ■■■■■■■
Suche Tauschpartner – Amiga 500!!!
Call: 08232/1065 Only 17–20 Uhr!!!
■■■■■■■■■■■■■■■■■■■■■■

Als Umsteiger in den DEHOCA!! PC- und Networkuser finden im Verband Public-Domain und jede Menge Tips zum Anwenden/Progr. Info: Postf. 1430, 3062 Bückeburg

Freesoft? Wo? Im Germany Public-Domain-Club ★ Info geg. Rückp. von GPDC-c/o Dieter Will, Postfach 2824, 2350 Neumünster, 04321/31711

Verkaufe Originale: Butcher V2.0 + Grabbit; jew. mit Anleitung je DM 35,—, Kohn W., Terassenweg 1, 8450 Amberg

A500 Top-Software A500
Spiele und Anwender
An: Markus Dohrmann
Schloßstr. 16, 2072 Bargteheide

Biete und tausche Amiga Software
Topaktuell!
Dieter Filsinger Flurweg 5
8670 Hof W.-Germany

Germany Public-Domain-Club ★ Info gegen Rückporto von GPDC- c/o Dieter Will, Postf. 2824 ★ 2350 Neumünster ★ 04321/31711

—— Suche Tauschpartner für Amiga ——
—— Suche Tauschpartner für Amiga ——
Auch Anfänger schreiben an: D. Bergrath, Wilhelmstr. 8, 5160 Düren (es eilt!!)

Amiga-Software
Tausche, verkaufe u. kaufe Software für Amiga 500/1000 — auch Anleit.! Call Michael: 07261/2865! Don't be foolish — Call us!!

Amiga im Haus C-64 Arbeitslos? Das muß nicht sein! Heizungsüberw. m. den C64-Rentner. Ausführl. Info geg. Freiumschlag von Thilo v. Stillfried, Meisenw. 13, 5983 Balve 1

Tausche Amiga und MS-DOS PD-Software Liste an Martin Mossakowski, Freytagstr. 28, 4730 Ahlen
(PS: Rechtlich einwandfrei)

SIDECAR für Amiga 1000, neu und original verpackt, 512 KB für DM 890,—!!! zu verkaufen. Ralf 0211/7976446 o. 746435

■■■ AMIGA-SOFTWARE ■■■
Write to Rainer Dietmer, Am sonnigen Hang 4, 6500 Mainz 32

Amiga Software! Call 04101/43254 Oliver

■■■ Originalsoftware ■■■■
Mindwalker (Spiel) Original mit Anl. 20,—
Textcraft 1.1 (Textverarb.) Original 20,—
Telefon: 0781/59130 ab 17.00 Uhr

Verkaufe: Original Shanghai 45,— DM.
Data Becker Bücher: Amiga 500 für Einsteiger 20,— DM, Supergrafik 40,— DM.
Tel. 07161/49463

Verkaufe org. DPaint 2, DVideo 2 und Defender of the Crown für 280,— DM.
Suche PD und andere Software zum Tausch
Tel. 069/703623

Als Amiga-Freak in den DEHOCA!! Ständige News in der PRINT und Amiga-AGs allerorten. Fast 70 Prozent aller Mitglieder sind Commodore-User. Info anfordern.

AMIGA **AMIGA**
Neueste Software
Tel.:
0291/3216

Habe 400 Disc P.D. und noch jede Menge andere Soft. Info anfordern bei Postfach 1151, 5067 Kürten. Gruß an alle Amiga Freaks! Kill the Virus!

Tauschpartner für Amiga gesucht
(auch Anfänger)
Tel. 0621/567514

Orig. m. Anl.: Worldgames, Wintergames, Kings Quest III je 35,— ★ 7 Cities of Gold (KS1.1) 30,— ★ Testdrive 50,— ★ Der Amiga (Sybex-Buch) 13,— ★ Zeitschr. 45 St. nur 60,— ★ 06622/2988 ab 15 h

Aegis Sonix, V2.0, mit zusätzlichen Instrumenten, VHB 110 DM. Amiga-Uhrenbausatz 50 DM. Jörg Marawski, Am Königsberg 6, 6239 Eppstein 3, 06198/32300

Verkaufe originale Bards Tale, One-on-One Marble Madness, Shanghai, Deep Space, Quiwi Mindwalker. à DM 40,— Tel. 0711/875957, Barthelmeß, N. Markgröninger Str. 42, 7 Stgt. 40

PD-Software! Info bei:
D. Komelter
Vom-Bruckplatz 45
4150 Krefeld

★★★ **Verkaufe Original Software** ★★★
Gegen Nachnahme: **Profimat** v. Data Becker für 49,— / **The C64 Emulator** für 59,—
T. Lüttich, Sartoriusstr. 20, 4300 Essen 1

Tausche und verkaufe Public-Domain Software für Amiga: Tel. 089/8418412 nach 14 Uhr

■■■■■■■■■■■■■■■■■■■■■■
Suche immer noch Tauschpartner!
Tel.: 0451/76769 ab 17 Uhr
■■■■■■■■■■■■■■■■■■■■■■

Verkaufe UBM-Text 2.2 Textverarbeitung Amiga, original m. Handbuch 90,— DM
Tel. 08122/15677

An alle Computerclubs!!!
Informiert euch über die Vorzüge als Ortsgruppe des DEHOCA — auch Einzelmitgliedschaften möglich. Info: Postf. 1430, 3062 Bückeburg

OK, here is the world famous sudl again: and he is searching for new contacts all over the world: Write to: Tom Seidel, Herrenstr. 24, 8360 Deggendorf: Answer 100%

Tausche Amiga PD-Software 2:1.
Martin Schlüter
Ringstraße 10
4358 Haltern-Hullern

Originale: Grabbit 30,—, Videoscape 3D 180,—, Deluxe Paint II 120,—, Sculpt 3D 100,—, Marble Madness 30,—, suche Druckertreiber für NEC-CP6 Tel.: 06152/63321 ab 17 Uhr

Verkaufe Original-Spiele: Starglider, Jagd auf Roter Oktober je DM 40,—, TV-Modulator DM 40,—, kaufe Original-Software. F. Geyer, Ringschnaiterstr. 42, 7955 Ochsenhausen 1

Amiga-Software ★ Amiga-Software ★ Amiga.
Auch PD. Immer auf dem neuesten Stand. Tel.: 0201/696293 ab 19 Uhr, nicht früher.
0201/696293 ★ Wega ★ 0201/696293

IBM und Amiga PD-Software (700 Disk) gegen Unkostenbeitrag abzugeben (4,—/Disk) incl. Disk. D. Senftl, Roßmarkt 30, 7300 Esslingen. Tel.: 0711/357908 (öfters probieren)

Verkaufe, alles Originale: Prism, Sculpt-3D, DPaint II, Digipaint, TV-Text 3D, Forms in Flight, CLI-Mate, Alfred Pasieka, Postfach 665, 4010 Hilden, Tel. 02103/41452

★★★ Hallo Freaks ★★★
Ich tausche, suche und biete immer neueste Software an. Auf Amiga und C'64!!
Ruft doch mal an Tel. 02181/499949

★★★ Top Software! ★★★
★★★ Tausche und verkaufe Software ★★★
★★★ Bitte anrufen Tel. 07131/77979 ★★★

Org. USA-Software, noch nicht benutzt:
DeluxePaint II, DeluxeVideo I je 190,— DM
DeluxePrint + Art Disk 1 für 150,— DM
Tel.: 0981/61111

■■■■■■■■■■■■■■■■■■■■■■
Starglider, Terrorpods je 30 DM beide = 50 DM! Originale! Tel.: 07741/2589
■■■■■■■■■■■■■■■■■■■■■■

★★★ **DEV-PAC-Assembler-** ★★★, nagelneu, nur 150,— DM /// außerdem: C64-Komplettsystem für MIDI-Freaks (C-Lab Scoretrack, Hitec-Interf...) 1300,— DM
A. Werner, 0711/244478

Atom-Animators
for new STUFF
0551/795804
Sven Meyer, Hauptstraße 72, 3400 Göttingen

Ausland

■■■■■■■■■■■■■■■■■■■■■■
Habe neueste C-64/Amigasoft! Verlangt Listen bei: D. Vogel, Friedmattstr., CH-6260 Reiden
■■■■■■■■■■■■■■■■■■■■■■

Tausche + verk. Amigasoftware aller Art. Schickt Liste: My Address: Auktje v/o Veem, Ammastrat 1a, 9724 LD Groningen, Nederland, Tel.: 050/142599

Suche Amiga Tauschpartner für Software aus aller Welt, 100% Antwort. Write to: Maryke Hoytink, Nieuwe-Pyramide 54, 3962 WW Wÿnk bÿ Duurstede, Holland

★ Du suchst Software für Deinen Amiga? ★ Software aller Art superpreisgünstig abzugeben! (Tausch?!) — Mario Sepp, Roseggerstr. 4, A-6020 Innsbruck Antw. 100%

Suche Tauschpartner für Amiga 1000! Habe viel Amiga Software zum Tauschen! Schickt Eure Liste an: Lessyn Thierry, 77 Rue des 3, Cantons Dippach-Gare Luxemburg!!

■ AMIGA-CS-AMIGA ■
■ Habe und tausche AMIGA Software ■
■ Schickt Disks und Liste an: Z. Krkoska ■
■Horni 117, CS-74401 Frenstat p.R., CSSR■

★★ Hello Computer-User and USA-Fans ★★
I always have the newest Stuff! Like: Gee Bee air Rally, California-Games aso.. Lists to NSD, Erlenweg 11, CH-8302 Kloten!

■■■■■ Newest Software ■■■■■
Call: 0041/036/451107 ★ CH ★
0041/036/451397 ★ CH ★
■■■■■■■■■■■■■■■■■■■■■■

★ Amiga ★ CH ★ Ludware ★ CH ★ Amiga
Verkaufe Leerdisks, sehr günstig. Sofort Preisliste anfordern bei: Daniel Luthy, Quellenstr. 6b, CH-8200 Schaffhausen

The Clan presents: AMIGA-SOFT in der CH. Topaktuelle Soft- & Hardware zu Spitzenpreisen. Info bei: Roger Wattenhofer, Buelstr. 7, 8854 Siebnen. Tel.: 055/642065

★★★ Österreich ★★★
Amiga Software wegen Hobbyaufgabe günstig abzugeben. Tel. 0222/3928583

Suche: Hardware

Wenn der Durchblick fehlt: Der DEHOCA-Service »Frageaktion« klärt auf im Zusammenwirken mit Firmen und Verlage. Info gibt's unter Postf. 1430, 3062 Bückeburg

Suche Amiga 500, 1000 o. 2000 günstig zu kaufen mit Monitor 1081 und Software. Angebote mit Preisvorst. bitte schriftl. an K. Daschowski, Zeppelinstr. 6, 4330 Mülheim

★ AMIGA 2000 Datenaustausch PC-Karte ★
Suche MS-DOS mit funktionsfähigen Dateien, JDISK, JLINK sowie Workbench ab 7.5.87, Oswald, Wielandstr. 14, 1 BLN 12, 030/3244249

Wer schenkt armen Schüler einen Amiga-Drucker oder Computer-Schrott?
Meldet Euch bei Michael Klimesch, Rossinistr. 10, 8070 Ingolstadt, Tel. 0841/86591

★★★ Kaufe AMIGA 2000 ★★★ Meldet Euch bei Resenhoeft, Tel. **05331/72872.** Suche auch billige 1541. Werden wir einig, hole ich den Computer selbst ab! Oder zahle Porto!

■■■■ Kaufe ■■■ Amiga ■■■■
Suche Amiga 1000 oder 500 mit Monitor 1081. Zahle etwa 1200 bzw. 1100 DM. Tel.
■■ 02822/53167 ■■ Suche Amiga ■■■

■■■■■■■■■■■■■■■■■■■■■■
Kaufe defekte Amigas 500 — 2000, C64er und Floppies z. Höchstpreisen, Tel. 04351/43670

Bitte helft einem armen Schüler, der verzweifelt irgendeinen **Amiga** mit oder ohne Zubehör sucht: M. Ellermeier, 5588 Blankenrath, **Tel.: 06545/442. Help!!! Amiga!**

★★★ Suche Dringend ★★★
Amiga 2000, XT-Karte, 24 Nadel Color Drucker (bes. NEC P 6), 5¼ Zoll Laufwerk, Festplatte, usw. Tel. 06043/2450 oder 05631/8360 (2. Nummer nur vormittags!!!!!)

Als Anfänger in den DEHOCA!!
Für 5 Mark im Monat Beitrag gibt es viele Vergünstigungen, Angebote und Kontakte — lokal und bundesweit. Postfach, 3062 Bückeburg

Suche Amiga 2000 ohne Monitor!
Tausche gegen Amiga 500 und 2 Videorecorder oder 1 Akai Stereoanlage und 1 Videorecorder, ruft mich an 06305/1433, ich lasse mit mir handeln

★ Wer schenkt o. verkauft Jugendzentrum ★
★ Amiga + Zubehör (auch def.) ★ Computerkurs? Haben wenig Geld! Tel.: 040/7225110 Mo. bis Mi. ab 16 Uhr (Marijan verlangen)

AMIGA goes family: —The Dallas Game— Public Domain BUS # 13: Leer+ 10 DM oder andere PD an Helmut Cordes, Nd. Stadtm. 17, 3549 Volkmarsen (frank. Rückumschlag!)

KAUFE # # KAUFE # # KAUFE #
Suche dringend Amiga 500, 1000 oder 2000, Monitor, Drucker, Laufwerk (3+5 Zoll!), Speichererw., Zub., usw. Tel. 06043/2450 o. 05631/8360 (2.Nr. nur vormittags!!!)

Der DEHOCA am Telefon: Wer noch mehr über Deutschlands größte Usergemeinschaft und seine Zielgruppen wissen will, wählt an Werktagen ab 16 Uhr 05722/26939

Ausland

Armer Amiga-Freak (Schüler) sucht A2000 mit Monitor 1081, möglichst günstig. Meldet Euch bei: Rico Dossenbach, Büchbrunnen, CH-8447 Dachsen, Tel. 053/27275

SUCHE: ★★★ AMIGA 1000 ★★★
Wenn möglich ohne Monitor.
Ronny Bender, Nelkenstr. 6
CH-8600 Dübendorf, Tel. 01/8217744

Biete an: Hardware

NEC-1036A-Zweitlaufwerk im Metallgehäuse, anschlußfertig, abschaltbar, für A500/1000/2000, Preis ca. 290,—, Tel. 0911/403681
G. Stock, Passauerstr. 356, 85 Nürnberg 30

AMIGA 500 incl. Speichererweiterung 1 MB & Uhr ca. 3 Mon. Garantie wegen A 2000 für 1200,— P.D. auf Wunsch. A. Willbränder, Heckenerstr. 16, 5469 Windhagen, T. 02645/4424

——— DEHOCA-Bundeswettbewerb ———
Wer schreibt das beste Arcadegame? Einsenden an DEHOCA-Zentrale bis zum 30.4.1988, die besten Games werden profess. vertrieben

DER DEHOCA, Deutschlands größte, schönste und vielseitigste Usergroup. Jeder hilft jedem, in der Gemeinschaft liegt die Stärke!!! Info: Postf. 1430, 3062 Bückeburg

AMIGA Sounddigitizer f. fast alle Samplerprg., Anschl. an Kopfhörerbuchse, 70 DM, J. Weinert, Münsterberger Str. 11, 2900 Oldenburg, Tel. 0441/62741

AMIGA Sounddigitizer 2, Mikroanschl., Klangregelung, f. fast alle Sampler Prg., 130 DM, J. Weinert, Münsterberger Str. 11, 2900 Oldenburg, Tel. 0441/62741

Amiga: 5¼" Diskdrive anschlußfertig zu verkaufen für VHB 300,—, verkaufe ferner Heimcomputer, Drucker, Microdive VHB 800,— Tel. 05741/1895

Spitzen-Preise: Tel.: 02823/1575
Monitor 1084 für 579,00 DM
Drucker NEC P 2200 für 948,00 DM
Disketten 2D 10er Pack!!! für 22,90 DM

S/W-Videokamera NEU mit Anschluß für DIGI-VIEW incl. Wechseloptik 1,6/16 mm DM 580,— A. Willbränder, Heckener-Str. 16, 5469 Windhagen, T. 02645/4424

Kienzle 6100 Comp.Anlage, 2 Wechsellaufw. + Drucker + Cassettenlaufw., Info unter Tel. 02151/20088, Preis VB 1500,— Selbstabhol.

Verkaufe original Speichererweiterung für A500 auf 1 MB für 240,— DM
Tel. 08232/1701

68020-Adapter (Piggy-Back-Board) für Amiga-1000 VB 1200,— oder Tausch gegen 5¼-LW od. Speichererweiterung, H. Idensen 3H1, Glünderstr. 3B, 0511/709559 ab 20 Uhr

Floppy-NEC zu verk. 250,— Tel. 0211/330449

Verkaufe AMIGA-500
VB 850 DM
Tel.: 02261/26410 oder 05021/7518

Verkaufe Sidecar + 32 MB Festplatte für nur DM 1800. NP über 2500 DM!! Orig. 2. Floppy 300 DM.
Alexander Stampf Tel. 089/3614927

Deluxe Print, Originaldiskette + Anleitung, 3 Monate alt, kaum benutzt, wegen Zeitmangel zu verkaufen. Preis: 99,— !!! Tel.: 09404/8261 tägl. ab 19.30 Uhr

■■■ Verkaufe AMIGA 1000, 512 KB, ■■■
■ dt. Tastatur, o. Monitor. VB 1200 DM ■
■ Anrufe bitte nur von 17-21 Uhr ■
■■■■■■ Tel. 089/904170 ■■■■■■

Verkaufe Amiga 500 und Monitor 1081/1084. Auch Einzelverkauf.
Telefon: 0531/16733 öfter versuchen

PSEUDO 2000er: A500-**1MB-Uhr**/2000er ähnl. Gehäuse/abges. Tastatur — aufrüstbar Zus. LW u. Festplatte (PC/AT-Karte über Adapter) 2 Mon. alt DM 1850,—, Info Tel. 04641/8940

★★★ Verkaufe: Amiga 1000 + Mon. 1081 + ★★★ NEC1036A + Speichererw. auf 1,5 MByte ★ 5 Mon./Top-Zustand, nur komplett: 2580 DM, ★★★ Telefon: 05232/71661 ★★★

SIDECAR mit 512 KB und VC 20 Prozessor für VB 995,— zu verkaufen.
Tel. 069/779111

Private Kleinanzeigen

Wer hat noch keinen Amiga 1000? Zu verkaufen Rechner ohne Monitor
VB 1200,—
Tel. 02151/20088

★★ VERKAUFE ★★
3½" Zweitlaufwerk für A1000. Commodore 1010-Gehäuse. 3 Monate alt. VB 250,—
Tel. 0906/91375 ab 14.00

Verkaufe 68020-Prozessor (16 MHz) bzw. 68881 (20 MHz). Thorsten Chmielus, Zedernweg 2, 6272 Oberseelbach, Tel. 06127/8559/3826 oder 0231/673868

AMIGA 500 neu (mit Garantie) für DM 800,— Original-Monitor für DM 500,—, zusätzlich Speichererweiterung für DM 210,—, 2 Joysticks und Software zu verkaufen. Tel. 06121/462189

Panasonic Drucker KX-P1091 500,— Speichererweiterung 256 K (Original) 100,— für Amiga 1000 zu verkaufen.
S. Loschge, Tel.: 0911/803377

Amiga 1000 (PAL, dt. Tastatur, 512 KB) + Monitor 1081 + 2. Laufwerk (NEC 1036a) + Handbücher + Software + M+T Amiga Handbuch zu verkaufen VB 2100,— Tel. 02204/56956

Doppelfloppy für Amiga, Slimline, abschaltbar, VB 550,— DM, Tel.: 04941/5523

Verkaufe Amiga 2000 für 1250,— DM Monitor 1081 für 550,— DM, 4 Monate Garan., PC-XT-KIT für 830,— DM, Tel. 04167/1713, neues 2. LW für Amiga 2000 265,— DM

Verk. Amiga 500 900 DM, Ass. Handbuch 55 DM, Amiga Monitor 680 DM, PG Handbuch 50 DM, MCC Pascal 150 DM, T. 07246/6340, ★★
★★ Data Becker Assembler 70 DM ★★

Verkaufe: Farbdrucker MPS 1500C. Neuwertig für DM 700. Space Battle, Two and Two, je 30 DM. **Suche:** DBW Render+Anleitung, suche billig Originale + PD-Soft. Tel.: 05661/8819

Verkaufe Amiga 1000 (PAL, dt. Tastatur, 512 KB) + Monitor 1081 + 2. Lw. (NEC 1036a) + Handbücher + Software. VB 2100,— DM. Tel. 02202/31511

■■■ AMIGA-2000 + 68020 ■■■
Der Knüller! 68020+68000 umschaltbar in einem Computer!! Info bei: Berthold Wagner, Akazienweg 10, 5000 Köln 50

Verkaufe Speichererw. A501 für Amiga 500 (1 MB inkl. Uhr) für 180 DM.
Tel. 05251/71381 ab 18 Uhr

Sidecar NEU für 999 DM zu verkaufen, 512 KB aufgerüstet incl. MSDOS 3.2 Tel. 06471/61119 ab 20.00 Uhr; suche deutsche Anltng. für Digi View 2.0 sowie Software etc.

Verkaufe Amiga-Monitor 1081, wegen Fehllieferung, unbenutzt, originalverpackt, Preis 590,— Näheres unter Tel. 09492/5023, Roland Knauer (ab 19.30 Uhr)

AMIGA 1000, m. Monitor, 2,5 MB, PAL, m. div. Software zu verk.
Tel. 06182/22116 abends

Verschiedenes

Wer schreibt das »DEHOCA-Spiel«? System egal, dem Sieger winkt ein PC-AT mit 40 MB und NEC-P 6 col., alle Mitglieder spielberechtigt! Info: Postf. 1430, 3062 Bückeburg

★ COMPUTERCLUB INTERNATIONAL ★
Der führende Club für alle Amiga- und ST-Freaks. Zeitschrift, PD-Soft, Beratung. CCI, Dorstener Str. 31, 4350 Recklinghausen. Tel.: 02361/15943 (Rückporto!)

Suche Kontakte zu AMIGAFREAKS! (Besonders zu Grafik- u. Soundbegeisterten) Am besten aus Frankfurt! Martyn Lawson, Julius-Leber-Weg 9, 6230 Frankfurt 80! (Habe kein Tel.)!

AMIGA-PROGRAMMIERER GESUCHT! Könnt Ihr in Amiga-Basic oder C Programmieren? Dann ruft an! Auch weniger geübte Leute! Tel. 089/1783101 ab 19.00

Einbau von int. Zweitlaufwerk in Amiga 2000 supergünstig. Wollen Sie sich lästige zeitaufwendige Selbstarbeit ersparen? Dann rufen Sie an: 069/593530 v. 16 bis 20 Uhr

Suche Kontakte zu netten AMIGA-Anwendern, auch Neulingen, zwecks Erfahrungs- und Programmtauschs. Ich nutze mein Gerät privat und geschäftlich. Michael Lange, Boppstr. 4, 1000 Berlin 61, Tel. 030/6931028

Private Kleinanzeigen

Verkaufe: Fernsehmodul A520Mod DM 30,—, Data-Becker-Bücher: Amiga für Einsteiger und Tips + Tricks zusam. DM 30,—
Tel. 08331/73437

Auch 1988 wieder auf mindestens 35 Messen! Spielewettbewerbe auf Hot-Labels namhafter Softwarehäuser. Starke Preise. Das ist der DEHOCA, Postf. 1430, 3062 Bückeburg

The Catastrophe now on Amiga: Disaster Area, searching for new contacts. Take the hotline and call: 069/350853

IGA ● Interessengemeinschaft Amiga ●
Der einzigartige Amigaclub ● kostenloses Info von IGA c/o Franz L. Mörsdorf, 7500 Karlsruhe 1, Waldenburger Str. 9

DRINGEND
Suche Amiga-Magazine Ausg. 6/7 und 8/9
Roland Kühn
Tel. 08232/2577 ab 18.30 Uhr

Ich grüße alle Amiga-User, besonders die der Bü 33 in der GFA HH. Kauf Dir auch einen, Uwe! Frohe Weihnachten ohne Schulstreß wünscht Euch allen Amiga-Ronny!

★ Wer programmiert in C oder Assembler ★
★ und möchte seine Erfahrungen mit mir ★
★ austauschen? Ruf doch einfach mal an ★
★ Norbert 0211/4380851 Düsseldorf ★

■■■■ Bücher ■■■■
Das AMIGA-Handbch (M+T) 30,—, Amiga Basic-Buch (tewi) siehe Ausgabe 12/87, Seite 36 35,—, Tips + Tricks (M+T) 30,— 0781/59130

SUCHE: Anleitung für AC-Basic-Compiler sowie Drucktreiberprogramm für NEC-CP 6 — Dringend! H. Podlucky, Eibenweg 6, 6085 Nauheim, Tel. 06152/63321 ab 17 Uhr

Suche Kontakte zu Amiga-User im Raum ME-Düsseldorf. Bin (Amiga-)Anfänger. Bes. an Erstellung eigener Software interessiert. Tel. 02104/74751 — Michael Stein

PC-Karte für Amiga 2000 plus 30 Disk MS-DOS Software für 1250,— DM zu verkaufen! O. Saenftl, Roßmarkt 30, 7300 Esslingen, Tel. 0711/357908 (öfters probieren!)

■■■ **Schluß mit dem Disk-Chaos** ■■■
■■■ Wir haben Disketiketten aus ■■■
■■■ Sammelbest. abzugeben. Info: ■■■
■■■ TOM, Pf. 1162, 4516 Bissendorf ■■■

■■■ SUCHE ■■■
■■ AMIGA-MAGAZIN ■■
Ausgaben 6/7 und 8/9
Tel. 09122/76926

DRINGEND
Suche Amiga-Magazin, Ausgabe 8/9
Tel. 06103/64714

Private Kleinanzeigen

■ Hey Freaks!!! We are searching for ■
■ contacts all over the world!!!! ■
■ Contact us: 04193/3456; West-G.!!! ■

Kreativteam sucht wortgewandte Gymnasiasten f. kl. Nebenjob. Wenig Zeitaufwand, aber evt. Courage erforderlich. Zuschr. an T. Jödicke, Prinzenstr. 42, 4459 Neuenhaus

Suche Kontakte zu netten Amiga-Freaks, Hauptinteresse Raytracing/Animationen in Raum Stuttgart. Tel. 0711/875957 ab 18.00 Uhr, N. Barthelmeß, Markgröninger Str. 42, 7 Stgt. 40

Der Schutzbrief kommt! DEHOCA und ein namhafter Versicherer sorgen für eine Vollkaskoversicherung auf Hardware — wie beim Auto!!! Info: Postf. 1430, 3062 Bückeburg

Suche erfahrenen A2000-Besitzer im Raum München, der mich beim Kauf eines A2000 berät und mir über die ersten Schwierigk. hilft. Corinna 089/1576881 ab 20 Uhr

Ausland

★★★ SWEDISH AMIGA USER ★★★
looking for German contacts!! Only news write to: TNO, Gullkragegatan 50, 230 42 Tygelsjö, Schweden, Tel.: 040466960 ...

Searching for contacts all over the Galaxy!!!
Spec. USA / GB / BRD / AUS
Write to: Thomas Bickec, Grubenstr. 9
CH-8544 Sulz-R'bach CMI

Gewerbliche Kleinanzeigen

AMIGA-500 Port-Expander — 4 Erweiterungen am Exp. Port — 78,— Tel.: 02226/5714
P. Engels, Kreisstr. 29, 5308 Rheinbach 14

AMIGA-DIGI-SERVICE, kostenlose Informat. anfordern, Demo-Disk DM 15,— (incl. bei Vorkasse/Scheck, Nachnahme + 4,50), K. Juris, Grafik, Bahnhofstr. 106, 6392 Neu-Anspach

PD-Software aus Deutschland zu Top-Preisen — Bei: T. Wirz, Höhenweg 98, 5300 Bonn 1

USER-PORT für Amiga-500 ★ 40 I/O Ports — 4 Timer ★ Steckkarte für Exp.Port 98,—, 30,— DM Leerpl., 02226/5714 Engels, Kreisstr. 29, 5308 Rheinbach 14

Wichtiger Hinweis:

Zur Bezahlung von Kleinanzeigen werden weiterhin keine Briefmarken angenommen

Achtung:

Wir machen unsere Inserenten darauf aufmerksam, daß das **Angebot**, der **Verkauf** oder die **Verbreitung** von urheberrechtlich geschützter Software nur für Originalprogramme erlaubt ist.

Das Herstellen, Anbieten, Verkaufen und Verbreiten von **»Raubkopien«** verstößt gegen das Urheberrechtsgesetz und kann straf- und zivilrechtlich verfolgt werden. Bei Verstößen muß mit Anwalts- und Gerichtskosten von über DM 1 000,— gerechnet werden.

Originalprogramme sind am Copyright-Hinweis und am Originalaufkleber des Datenträgers (Diskette oder Kassette) zu erkennen und normalerweise originalverpackt. Mit dem Kauf von Raubkopien erwirbt der Käufer auch kein Nutzungsrecht und geht das Risiko einer jederzeitigen Beschlagnahmung ein.

Wir bitten unsere Leser in deren eigenem Interesse, Raubkopien von Original-Software weder anzubieten, zu verkaufen noch zu verbreiten. Erziehungsberechtigte haften für ihre Kinder.

Der Verlag wird in Zukunft keine Anzeigen mehr veröffentlichen, die darauf schließen lassen, daß Raubkopien angeboten werden.

Gewerbliche Kleinanzeigen

SUPER-FARBFOTOS von AMIGA-IFF-Grafiken, hochauflösende Color-Fotoabzüge in allen Größen sowie Visitenkarten fertigen wir in kürzester Zeit von Ihren auf Diskette gespeicherten Grafiken. Info v. Optivision, Heckenerstr. 16, 5469 Windhagen, Tel.: 02645/4424

DRUCKER umstecken? Nie mehr — autom. Schalter 2 RECHNER-1 DRUCKER — kontaktlos. Fertig: 78,— BS: 58,—, PL: 22,—, 02226/5714 Engels, Kreisstr. 29, 5308 Rheinbach 14

Computer, Drucker, Monitore, Terminals, Programme und Zubehör für Amiga und PC
Jürgen Reichardt-Kron, Hard & Software, Oberdorfstr. 12, 6395 Weilrod 3

Thai Boxing, Grid Start, Vader je 33,— DM
Fa. Fred Martschin, Postf. 142, 3258 Aerzen

Biete an: Public Domain ab 3,50 DM bis 3,70 pro Disk je nach Abnahme, 3 Infodisketten unter 02327/89293 anfordern in der Zeit von 7.30 bis 18.30 Uhr

FAKTU-PLUS V1.21 DM 49,— u. 5,— Porto/ Verp. Fakturierungsprogr. mit Kunden-/Artikelverw. Rechn.-/Mahnwesen, Etikettendruck, uvm. KD-Software, 5100 Aachen, Drosselweg 87a

Public Domain (FISH u.a.) 10 Disks 45 DM, 20 Disks 85 DM + 3 DM Porto bzw. 5 DM NN.
P. Keim, Vogelsanger Str. 34, 5 Köln 30

»Endlich! **Prof. Kurvendiskussionsprg.** für alle Amigas! **Nathan d. Weise** findet Nullstellen, Wende-, Extrempunkte und Abl. aller Funktionen (auch gebr.rat.). Er kann Integration, Regression, berechnet Rotationskörper + Gleichungssyst. Info für frank. Freiumschl.! Nur 99 DM (+NN 8 DM/Scheck + 5DM)! **EINSTEIN-SYSTEMS,** Thomas Klucke, Bunsenstr. 6e, 6200 Wiesbaden, Tel. 06121/600172«

★ KLEKS 32 ★ Version 1.0, Amiga ab 512 K Das Super BASIC-Malprogramm. Sehr schnell Interessant für alle Basicfans, da vom Benutzer selbst um eigene Funktionen erweiterbar. Viele Funktionen, wie Lupe, Füllmuster, etc. Laden und Speichern im IFF Format, somit kompatibel zu Graphicraft etc. Mit deutscher Anleitung. Bei Vorkasse in bar nur 30,— DM. Bei NN + -Geb. Zu beziehen direkt vom Autor. M. Uhlir, Waldstr. 43, 4230 WESEL

AMIGA 500/2000 Hard- und Software
Aus unserem Angebot Spiele: Leviathan 69,—, Bad Cat 59,—, Garrison 65,— u.v.m.
3½" Zweitfloppy für A 500, kompl. 369,— DM
als 3½" Einbaufloppy für A 2000 299,— DM
Fordern Sie unsere kostenlose Liste an!
KNACK Computertechnik
Wilmsmannstr. 14, 4600 Dortmund 30

Amiga-Chip 8520A (CIA) nur DM 49,45! + NN-Versand DM 7,50; CIK-Computertechnik
Ingo Klepsch, Tel. 02333/80202

Disketten NoName 3½" 1DD-DM 24,50
Disketten NoName 3½" 2DD-DM 26,50
AMIGAS-DRUCKER-ZUBEH.
ELEKTR. BAUTEILE
Preisliste anfordern: W. Niemann
Walsumermarkstr. 140, 42 Oberhausen 11

AMIGA 500 und 2000 Hard- und Software
Aus unserem Angebot: Digi View 297,—, Digi Paint 98,—, Videoscape 3D 279,—, King of Chicago 89,—, Gunship 89,— u.v.m. 3½" Zweitfloppy für A 500, kompl. 359,—
als 3½" Einbaufloppy für A 2000 289,—
Stereo-Einbaukit für Monitor 1081/4 79,—
Fordern Sie unsere kostenlose Liste an!
KNACK Computertechnik
Wilmsmannstr. 14, 4600 Dortmund 30

— AIT-USER-GROUP —

— Vorsicht — Computer-Viren —

Möchten Sie sich davor schützen oder Ihren Virus killen?

Wir haben den SCA-Virus-Protector und SCA-Virus-Finder.

Für nur 5,— DM pro Disk plus Porto, wie alle von über 500 Public-Domain-Disks. Super-Info-Disks = 2 x Disk = 12,— DM = >1300 KB >550 Screens.
Berechtigt zum Tausch von 4 zu 1.

— AIT-TEAM, M. RÖNN
Ziegeleiweg 32, 3257 Springe 4
Telefon 05041/8229

Die erste AMIGA-Extra

Programme der Extraklasse, die zu lang für den Abdruck im AMIGA-Magazin sind, kommen ab sofort auf die AMIGA-Extra-Diskette. Dort finden Sie dann mehrere gute Programme, die jeder Amiga-Besitzer einfach haben muß.

Auf unserer ersten Extra-Diskette befinden sich diesmal drei Programme der Spitzenklasse: »CADos3D«, »Fractal Construction Kit« und der »Funktionsplotter«.

CADos3D (Bild 1) ist ein Programm mit dem leicht dreidimensionale Körper entwickelt und gedreht werden können. Es arbeitet im hochauflösenden Modus (640 x 512 Punkte). Die Steuerung erfolgt bis auf die Eingaben von Texten durch die Maus.

Die Eingabe von beliebigen Körpern mit CADos3D ist denkbar einfach. Es werden Punkte festgelegt, die das Programm mit Linien verbindet. Diese Methode hat viele Vorteile, so sind zum Beispiel nachträglich einzelne Punkte verschiebbar. Die entsprechenden Linien verändern sich dann automatisch.

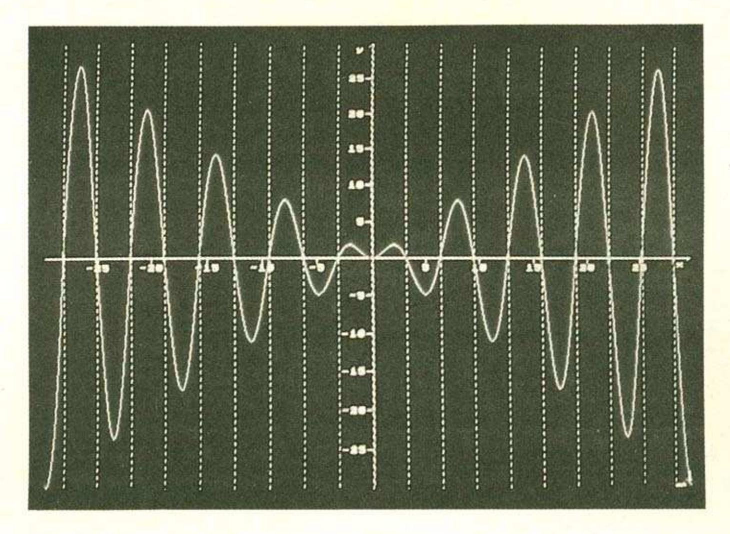


Bild 3. Eine große Hilfe beim Auswerten von mathematischen Funktionen stellt der »Funktionsplotter« dar

Bild 1. Komfortable Erstellung von 3D-Drahtgittermodellen und deren Drehung sind mit »CADos3D« kein Problem. Das Programm »Move« erlaubt später die Vorführung einer Bewegung mit Text auf der rechten Seite.

Der Körper läßt sich auch in den drei verschiedenen Ansichten (von links, von vorne und von unten) darstellen. Durch Anklicken eines einfachen Schalters bringt ein angeschlossener Drucker dieses Bild dann zu Papier.

Durch Schieberegler bestimmt, kann man nun den Körper beliebig im Raum drehen lassen. Dies geschieht entweder schrittweise durch Mausklick oder stufenlos.

Doch nun zum Besonderen an CADos3D. Dort wo die Schalter auf dem Bildschirm zu sehen sind, kann man Texte darstellen. Der Körper wird mit den voreingestellten Parametern gedreht und rechts ist der Text zu sehen. Genauso sieht es später aus, wenn Sie mit dem Zusatzprogramm MOVE arbeiten. Dieses Programm läßt nämlich die mit CADos3D generierte Animation ablaufen.

Das nächste Programm generiert, wie der Name Fractal Construction Kit schon sagt, Apfelmännchen, Juliamengen, Feigenbaum-Strukturen und die dazugehörigen Verhulst-Diagramme.

Durch die Verwendung von Assembler und die schnellen Fast-Floating-Point-Routinen für die zeitkritischen Prozesse und einige Tricks gehört das Programm zu den schnellsten dieser Art.

Das Programm wird über ein Hauptmenü und die Tastatur bedient. Was zunächst als Nachteil erscheint, ist in Wahrheit ein Vorteil. Da die Maus nicht verwendet wird, kann auf die Intuition-Bibliothek verzichtet werden. Dadurch aber wird das Programm schneller, als mit Mausbenutzung.

Die Bildschirmdarstellung läßt sich problemlos verändern. Unterstützt werden dabei in der Horizontalen 320 oder 640 Punkte beziehungsweise 200 und 400 Punkte vertikal. Es stehen dann 32 oder 16 Farben zur Verfügung.

Die errechneten Bilder lassen sich nach der Fertigstellung im IFF-Format speichern. Dadurch können sie jederzeit mit Zeichenprogrammen wie Deluxe Paint oder Graphicraft weiterbearbeitet werden.

Vor allem aber beeindruckt die Geschwindigkeit von Fractal Construction Kit. Das erste Apfelmännchen im Lores-Modus und 31 Iterationen benötigt nämlich nur 3,5 Minuten.

Das letzte Programm im Bund ist der Funktionsplotter. Dieses komplett in Assembler geschriebene Programm erlaubt die schnelle Überprüfung und Darstellung auch von komplexen Funktionen. Das Programm arbeitet mit der Auflösung 640 x 512 Punkten, die eine genaue Zeichnung zuläßt. Bedient wird das Programm über Pull-Down-Menüs.

Nach dem Eingeben der gewünschten Funktion, wird diese innerhalb von Sekunden auf den Bildschirm gebracht. Auch bei sehr komplexen Funktionen liegt die Rechenzeit praktisch nie über 10 Sekunden.

Aber die Zeichnung der Funktion allein, bringt meist recht wenig. Darum bietet das Programm noch einige interessante Fähigkeiten. Gerade für Schüler wichtig ist die Berechnung der Nullstellen einer Funktion. Auf dieselbe Weise erfolgt auch das Eintragen der Minima und Maxima der Funktion.

Falls Definitionslücken existieren (zum Beispiel bei der Wurzel, wenn x kleiner 0 ist), findet der Funktionsplotter auch diese und Sie können den Bereich markieren lassen.

Alles in allem ist es ein sehr nützliches Programm für diejenigen, die sich (freiwillig oder zwangsläufig) mit Mathematik beschäftigen. (rb)

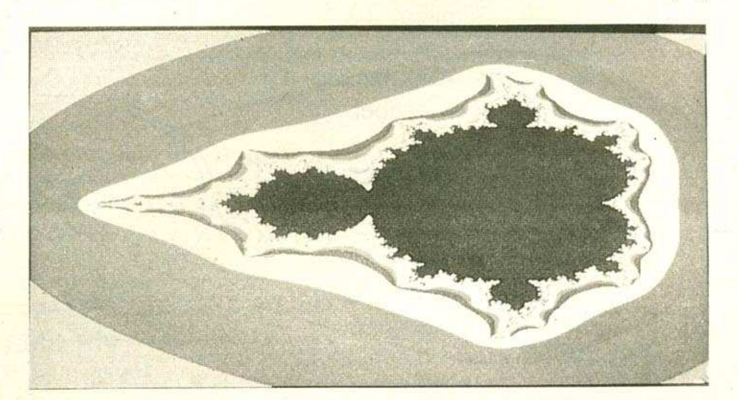


Bild 2. Ein »Muß« für Liebhaber von Apfelmännchen ist das »Fractal Construction Kit« auf der AMIGA-Extra-Diskette

Der Gipfel: Maxiplan Plus

AMIGA test

Halten Sie auch das Arbeiten mit Kalkulationsprogrammen für umständlich? Dann vergessen Sie am besten alles, was Sie über »Spreadsheets« auf anderen Computern wissen und lesen diesen Test.

Die Meinung über Kalkulationsprogramme ist durchaus nicht einstimmig. Diejenigen Anwender, die sich in die Struktur und Arbeitsweise solcher Programme eingefunden haben, schätzen »Spreadsheets« als nützliches Hilfsmittel. Die Gegner bemängeln deren umständliche Bedienung.

Durch den Amiga oder andere Computer mit einer derart komfortablen Benutzeroberfläche ist eine neue Generation von Softwareprodukten möglich und — mit Maxiplan Plus auf dem Sektor Kalkulation — bereits auf dem Vormarsch. Dieser Test soll nicht nur die Leser ansprechen, die sich ohnehin schon mit dieser Thematik auseinandersetzen. Er zeigt Anwendungen auf, die sich bei der Bewältigung einer Vielzahl wirtschaftlicher Probleme sinnvoll einsetzen lassen.

Maxiplan Plus ist bisher nur in der englischen Version erhältlich. Nach dem Start des bootfähigen, nicht kopiergeschützten Programms stehen dem Anwender vorerst nur zwei Menüs zur Verfügung. Durch die Unterpunkte des Menüs »Control« lassen sich neue Arbeitsblätter öffnen, gespeicherte laden oder nicht mehr benötigte schließen. Die Anzahl der geöffneten Arbeitsblätter ist auf drei beschränkt.

Mit den Unterpunkten »Color Selection« und »Maximum Window Size« läßt sich die Anzahl der verfügbaren Farben zwischen vier und acht und die Anzahl der auf einem Bild dargestellten Zeilen zwischen 26 und 58 umschalten. Diese Einrichtung ist sinnvoll, um einerseits eine gute optische Gestaltung von möglichst viel Informationen zu erhalten, und andererseits den vom Bild genutzten Speicherbereich auf ein Minimum zu begrenzen.

Die Funktion »Open Workbench« bringt den Workbench-Bildschirm in den Vordergrund. Hier können nun andere Programme ausgeführt werden, ohne Maxiplan Plus zu verlassen. Die Anwahl von »Close Workbench« schließt die Workbench wieder. Da man ohnehin problemlos zwischen Workbench und Programm umschalten kann, hat die letzte Funktion nur dann einen Sinn, wenn der durch die Workbench genutzte Speicherplatz wieder freigegeben wird. Das funktioniert aber leider nicht.

Der letzte der interessanten Menüpunkte ist »Fast Cell Display«. Diese Funktion beschleunigt die Berechnung eines Arbeitsblattes nach Änderung einzelner Zellen. Dieser Modus ist schaltbar, weil er mehr Speicherplatz benötigt.

Ade Preferences

Mit den Funktionen des Menüs »Printer Control« hat der Anwender indirekt Zugriff auf die Systemkonfiguration. Das bedeutet, er benötigt für wichtige Druckereinstellungen nicht mehr das Programm Preferences. Problemlos lassen sich die Zeichendichten Pica, Elite und Fine, linker und rechter Rand, Seitenlänge, Zeilenabstand und die Druckqualitäten NLQ und Draft auswählen.

Der geübte Bediener von Amiga-Programmen benötigt nur einige Augenblicke, um un-

ter Nichtbeachtung des Handbuchs mit »New Worksheet« den ersten Schritt zu einem funktionsfähigen Kalkulationsblatt zu machen.

Schauen wir uns vor den ersten Experimenten den Arbeitsbildschirm (Bild 1) genauer an. Die Titelleiste des Programms enthält neben Angaben über den verfügbaren Speicher (Chip-RAM und gesamte Kapazität) auch das aktuelle Datum mit Tagesname und Uhrzeit. Zu jedem Arbeitsblatt gehört ein Fenster. Die Titelleiste des Fenster beinhaltet den Namen derjenigen Datei, unter der das Arbeitsblatt auf Diskette gespeichert ist.

Der zu sehende Ausschnitt des eigentlichen Kalkulationsblattes besteht aus 25 numerierten Zeilen und sieben mit den ersten Buchstaben des Alphabetes bezeichneten Spalten. Der maximal mögliche Umfang eines Kalkulationsblattes beträgt 32760 Zeilen x 512 Spalten. Nicht sichtbare Teile eines Arbeitsblattes können komfortabel mit zwei Rollbalken und vier Rollpfeilen sowohl vertikal als auch horizontal verschoben werden. Maxiplan nutzt übrigens die volle Bildschirmhöhe mit 256 beziehungsweise 512 Zeilen.

Da alle Theorie bekanntlich grau ist, wollen wir die weiteren Funktionen von Maxiplan Plus an einem Praxisbeispiel, einer einfachen Ein-/Ausgabenrechnung, erläutern (Bild 1). Mit der Maus bewegen wir den Mauszeiger auf das Feld B2. Nach Eintippen des ersten Buchstaben unseres Titels tauscht Maxiplan das Wort »Contents« (Inhalt) gegen »Editing« aus — wir können nun in dieser Zeile unseren Titel schreiben. Alle Zeichen von »Ein-/Ausgabenrechnung« sind Inhalt von B2 und obwohl dieser Titel länger ist als die Breite der Zelle, ist er in voller Länge auf dem Bildschirm zu sehen. Die Zellen C2 bis F2 bleiben leer.

Die Eingabe der Posten in Spalte A1 erleichtert Maxiplan Plus dadurch, daß sich der Eingabecursor nach jedem Return eine Zeile tiefer in derselben Spalte positioniert. Nach der Eingabe der Spaltenbezeichnungen (Q1, Q2, ...) positionieren wir den Mauszeiger auf die Zelle C5, drücken die linke Maustaste, halten diese fest und bewegen den Mauszeiger auf die Zelle G5. Dort lassen wir die Taste los. Mit dieser Aktion haben wir einen Bereich (englisch: Range) definiert, den Maxiplan Plus farblich gekennzeichnet hat. Größere Bereiche können auch mit einem Mausklick an der linken oberen Ecke und einen weiteren Klick bei gleichzeitig gedrückter Shift-Taste an der rechten unteren Ecke des Bereichs festgelegt werden. Die Definition eines Bereiches ist immer sinnvoll, wenn sich eine Aktion nicht nur auf eine, sondern mehrere Zellen auswirken soll. Im Menü »Format«, Unterpunkt »Alignment« (Ausrichtung), finden wir mit »right« auch die beabsichtigte Aktion: alle Zeichen der Zellen innerhalb des Bereichs werden rechtsbündig formatiert.

In die Spalte C werden die Verkaufserlöse, Einkaufspreise und Kosten für das Quartal 1 eingetragen. Die Zifferneingabe kann sowohl über die Tastatur als auch die Ziffernschalter oberhalb der Inhaltszeile erfolgen. Nun positionieren wir den Cursor auf die Zelle C17 und klicken die Schalter »=« und »f()« an. Am Zeichen »=« erkennt Maxiplan eine Formel. Der Inhalt von Formelzellen wird durch Berechnung oder Übertragung von anderen Zellen bestimmt. Bei Anwahl der Menüfunktion »Options Show« zeigt Multiplan Plus übrigens in den Formelzellen statt den Werten die entsprechende Formel.

Mit der Anwahl von »f( )« öffnet sich ein Fenster mit einer Liste der zur Auswahl stehenden Funktionen. Da nicht alle der 70 Maxiplan-Funktionen in das Fenster passen, kann mit einem Schieberegler der gesamte Funktionsvorrat auf- und abwärts gerollt werden. Wir wählen die Funktion SUM und nach einem Mausklick auf diese erscheint der Name und ein Hinweis über die notwendigen Zusatzangaben in den unteren Zeilen des Fensters.

In der Eingabezeile steht nun »=SUM( )«. Der Eingabecursor befindet sich auf der zweiten Klammer. Die erforderliche Zusatzangabe für die Funktion muß ein Zellenbereich sein. Also definieren wir diesen mit dem bereits beschriebenen Verfahren. Die

Eingabezeile enthält nun den Ausdruck »=SUM(C14:C15)«. Mit <Return> beenden wir die Eingabe — in der Zelle C17 erscheint nun mit 40000 das Ergebnis der Berechnung.

Mit »=C7-C8« für C10, »=C10-C17« für C19 und »=C18-C19« für C21 machen wir dies auch. Vorher haben wir allerdings die Zelle C19 mit der Formel »=C18*$D24« (20 Prozent Einkommenssteuer) belegt. Die Zellenangaben in den einzelnen Ausdrücken können übrigens durch Anklicken der jeweiligen Zellen erzeugt werden.

Formeln kopieren

Unsere Auswertung für das erste Quartal ist damit abgeschlossen. Als nächstes tragen wir die variablen Daten für die anderen Quartale ein und kommen dann zu einer Funktion, welche die Mächtigkeit solcher Kalkulationsprogramme beweist. Mit ihr vereinfacht sich die Prozedur der Summen- und Differenzbildung für die restlichen drei Spalten enorm. Obwohl alle Zellenangaben in den Formeln absolut angegeben sind, speichert Maxiplan diese ohne weitere Angabe relativ. Die Formel »=C7-C8« für C10 ist für das Programm die Anweisung, die Differenz aus den Zellen C(10-3) und C(10-2) beziehungsweise bezogene Zelle minus 3 und bezogene Zelle minus 2 zu bilden. Wir brauchen diese Formel also nur auf die anderen Zellen (D10, E10 und F10) zu übertragen. Das geschieht für den Rohgewinn mit der Bildung eines Bereichs von C10 bis F10 und dem Menüpunkt »Commands«, Unterpunkt »Fill« und »right«. Damit sich die Formel aus C19 (C18/100*$D24) beim Kopieren für D19 nicht in »D18/100*E24« ändert, enthält die Zellbezeichnung »$D24« den Prefix »$«. Damit ist diese Zellreferenz nicht relativ, sondern absolut.

Nachdem wir sämtliche Formeln auch auf die anderen Quartale übertragen haben, bilden wir für G7 (Verkauf Jahr) die Summe aus C7 bis F7. Anschließend kopieren wir diese Formel mit »Commands Fill down« in alle Zellen von G8 bis G21. Dadurch erhalten wir allerdings auch die Summe 0 für alle Leerzeilen. Mit einem Mausklick, <Del> und <Return> läßt sich jede einzelne dieser »Fehlsummen« leicht wieder entfernen.

Die leistungsstärkste Funktion von Kalkulationsprogrammen ist die Ausführung eines sogenannten »Was wäre wenn ..?«. Was wäre, wenn die Einkommenssteuer um 10 Prozent steigt? Wir tragen in Zelle D24 den Wert 30 ein und Maxiplan kalkuliert automatisch alle Formel-Verknüpfungen nach. Fast ohne zeitliche Verzögerung sehen wir die Auswirkungen auf das Jahreseinkommen. Die automatische Kalkulation kann abgeschaltet und durch eine manuelle ersetzt werden. Das verhindert während der Veränderung größerer Blätter unnötige Zwangspausen. An diesem einfachen Beispiel erkennen Sie schon die Mächtigkeit solcher Programme bei komplizierteren Kalkulationsschemata.

Schwarz auf weiß

Natürlich wollen wir unsere Arbeit nun schwarz auf weiß in den Händen halten. Kein Problem — die Menüfunktion »Project Print« anwählen und schon beginnt der Drucker mit der Ausgabe. Soll nur ein Ausschnitt des Arbeitsblattes gedruckt werden, so muß dieser erst als Bereich definiert werden. Mit »Project Print Setup« kann das ausgedruckte Kalkulationsblatt mit Zusatzinformationen versehen werden. Solche zusätzlichen Angaben sind etwa eine Seitennumerierung, das Datum mit der Uhrzeit, der Dateiname oder ein gesondert einzugebender Titel zu Beginn jeder Seite, die Bezeichnung der Zeilen und Spalten oder ein Strichraster für die Trennung der einzelnen Zellen. Die Elemente lassen sich durch Einzelanwahl oder Mehrfachanwahl (alle gewünschten Menüpunkte mit beiden Maustasten gedrückt überstreichen) einschalten. Die Print-Funktion überprüft bei jeder Ausgabe die Schalter und berücksichtigt die Einstellungen entsprechend.

Maxiplan Plus besitzt eine Reihe von Formatfunktionen für Zellen(bereiche). So kann der Anwender etwa bestimmen, ob der Inhalt einer Zelle zentriert, rechts- oder linksbündig, kursiv oder unterstrichen, fett oder farbig sein soll. Mit »Palette« läßt sich die Standardfarbeinstellung jederzeit den individuellen Bedürfnissen anpassen.

Paßt ein Rechenergebnis nicht mehr in eine Zelle, wird die Zelle mit »#« aufgefüllt. Mit »Standard«, »Wide« und »Specify« kann daher die Spaltenbreite auf 10 oder 67 Zeichen festgelegt beziehungsweise frei bestimmt werden. Die Anwahl von »Currency« (Währung) bewirkt das Setzen eines Währungssymbols in die Zelle. Der Anwender kann sowohl das Symbol als auch die Plazierung vor und/oder hinter dem Betrag bestimmen.

Eingaben in Zellen, die mit den Attributen »Time« und »Date« belegt sind, wandelt Maxiplan Plus in die entsprechenden Angaben um. Unangenehm für den deutschen Anwender fallen hier die englischen Abkürzungen für die Monatsnamen sowie die Angaben »AM« und »PM« für die Zeit auf. Das Attribut »Percent« ergänzt Beträge mit einem Prozentzeichen. Warum das Programm bei einer Anwahl dieser Funktion für Feld D24 (Einkommensteuer) aus der »20« den Ausdruck »2000.00%« macht, konnte nicht geklärt werden.

Nicht selten muß ein Arbeitsblatt mit vertraulichen Informationen an einen Mitarbeiter zur Bearbeitung weitergegeben werden. Solche kritischen Informationen lassen sich mit einem Paßwort sichern. Derart geschützte Zellen sind mit »-« gefüllt. Ihren Inhalt zeigt Maxiplan erst nach der Eingabe des Paßwortes. Für einen ähnlichen Zweck lassen sich Zellen mit »Protect« vor dem Überschreiben schützen.

Maxiplan Plus bietet dem Anwender für das Erstellen eines Arbeitsblattes sinnvolle Editierfunktionen. Mit »Cut« und »Copy« kann der Inhalt von Zellen oder Bereichen ins Clipboard, ein vom System zur Verfügung gestellter Zwischenspeicher, abgelegt werden. »Cut« löscht zusätzlich die angegebene(n) Zelle(n). Durch »Paste« und »Paste Data« läßt sich der Clipboard-Inhalt komplett oder ohne Berücksichtigung der Formatinformationen wieder in das Arbeitsblatt übernehmen. »Clear« und »Clear Data« löschen wie Cut die Zellen, der Inhalt des Clipboards wird jedoch nicht geändert.

Formelausdrücke setzen sich aus Operanden (bezogene Zellen, Konstanten oder Funktionsergebnisse) und Operatoren (Rechenzeichen) zusammen. Neben den Grundrechenarten einschließlich Exponentialrechnung stellt Maxiplan Plus auch logische (Vergleichs)Operatoren (=, <, >, <=, >=, <>) zur Verfügung. Das Ergebnis von Vergleichsoperationen ist 0, wenn der Vergleich eine Falschaussage ergibt, 1 bei einer wahren Aussage (der Vergleich »D26<30« ist für unser Beispiel wahr). Die Operandenhierarchie kann natürlich durch das Setzen von Klammern geändert werden.

Um die Rechenzeiten möglichst kurz zu halten, kann die Aufteilung eines komplexen Kalkulationsblattes in mehrere einzelne Blätter sinnvoll sein. So lassen sich etwa für jede unserer »Filialen« ähnliche Ein-/Ausgabenrechnungen aufstellen. Auf einem weiteren Arbeitsblatt können dann in der Zusammenfassung die Erlöse und Kosten pro Jahr der einzelnen Filialen aufgezeigt werden. Um die einzelnen Werte nicht manuell übernehmen zu müssen, bietet Maxiplan Plus die Funktion »REFER«. Diese Funktion benötigt zwei Argumente. Der erste ist der komplette Pfadname der Datei, unter dem das betreffende Kalkulationsblatt gespeichert ist, und der zweite die Zelle, aus der der Wert übernommen werden soll. Da die Zellbezeichnungen in der Regel schwer zu merken sind, kann eine Zelle oder ein Zellenbereich mit einem Namen versehen werden.

Gehörten die meisten der bisher dargestellten Funktionen von Maxiplan durchaus zum Standard, so entfaltet das Programm bei der grafischen Darstellung erst seine volle Leistungsfähigkeit. Mit den Daten der Arbeitsblätter lassen sich Linien-, Kuchen-, Balken-, Flächen-, XY-, Stufen- und Hoch/Tief-Diagramme anfertigen. Dreidimensionale Darstellung ist bei den Kuchen- und Balkendiagrammen (Bild 2) möglich. Die Lage des Koordinatensystems im Raum kann bei der 3D-Darstellung mit der Maus verändert werden. Flächendiagramme ähneln Liniendiagrammen (Bild 2). Allerdings ist die Fläche unter der Linie ausgefüllt. Ein XY-Programm ist die Darstellung von Koordinatenpaaren. Damit lassen sich komplexe Muster (Shapes) erzeugen. Im Gegensatz zu einem Liniendiagramm wird bei einem Stufendiagramm die Differenz zweier benachbarter Punkte durch entsprechende Stufen gekennzeichnet. Ein Einsatz für Hoch/Tiefdiagramme (Bild 3) wäre zum Beispiel der Kursverlauf von Aktien (Höchst-, Tiefst- und bei der Darstellung mit drei Variablen auch der Abschlußkurs).

Wahlweise können Grafiken mit Abszissen- und Ordinatenachsen, Grid, Achsenbema-

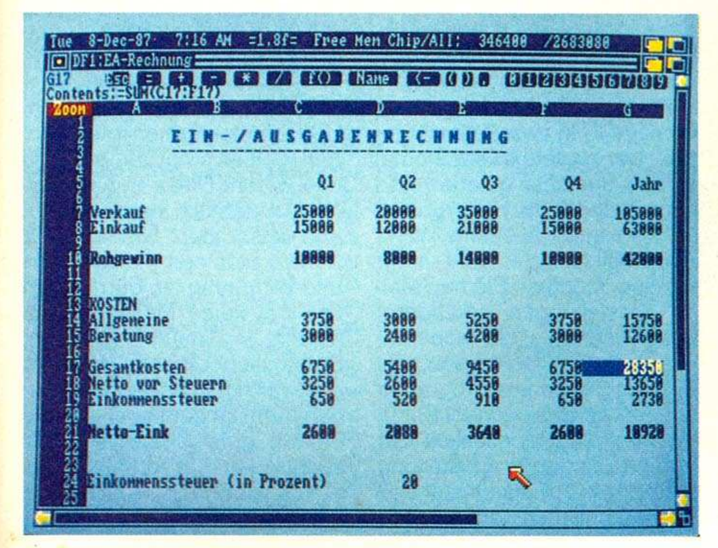


Bild 1. Der Arbeitsbildschirm von »Maxiplan Plus«

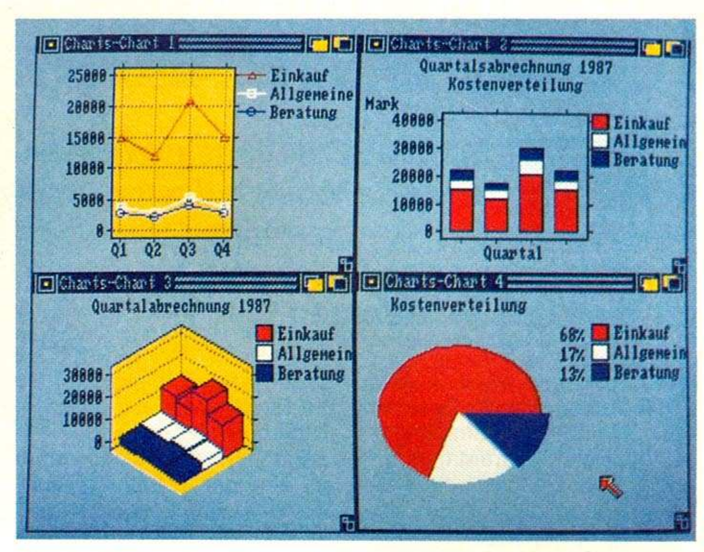


Bild 2. Werte können auch grafisch angezeigt werden

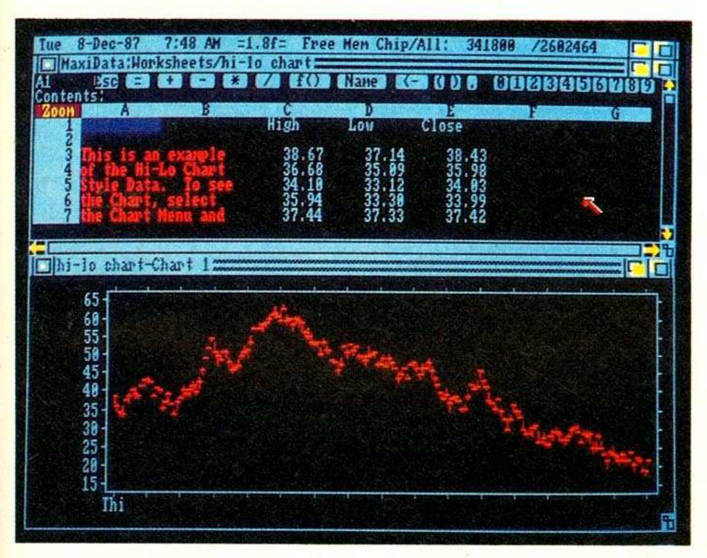


Bild 3. Ein Beispiel für ein Hi-/Lo-Diagramm

ßungen, Liniensymbolen (Bild 2 links oben), bis zu zwei Titeln und Achsenbezeichnungen (Bild 2 rechts oben) versehen werden. Die Skalierung gibt Maxiplan Plus vor. Mit der Maus können Linien- und Balkendiagramme verändert werden. Diese Modifikationen passen »auch« die Daten des Arbeitsblattes an. Durch Anklicken eines Diagrammelements läßt sich bei den meisten Diagrammtypen der Wert, der hinter diesem Element steckt, in der Menüleiste anzeigen.

Wie einige andere Kalkulationsprogramme besitzt auch Maxiplan integrierte Dateiverwaltungsfunktionen. Für die Definition einer Datei müssen lediglich die Datensätze in eine ununterbrochene Zeilenfolge eingetragen werden. Jede Zeile enthält ein Datenfeld. Die erste Zeile der Datei enthält die Feldnamen. Nach der Markierung und Benennung dieses Bereichs und der Ausführung von »Define Database« kennt Maxiplan Plus eine (weitere) von 63 möglichen Dateien pro Arbeitsblatt. Auf die gleiche Weise werden bis zu 63 verschiedene Kriterien für Such-, Lösch-, Sortier- oder Extrahiervorgänge definiert. Mit der Funktion »Extrahieren« lassen sich Datensätze, die einem bestimmten Kriterium entsprechen (Beispiel: Name = Meier), in einen anderen Teil des Arbeitsblattes kopieren.

Maxiplan Plus kann Dateien als ASCII-Text speichern beziehungsweise laden. Das erleichtert die Weiterverarbeitung der Daten beispielsweise mit Textverarbeitungsprogrammen. Dateifunktionen unterstützen unter anderem die Summenbildung, die Bestimmung von Minima/Maxima und statistische Auswertungen von Datensätzen.

Jeder Arbeitsschritt während der Bedienung von Maxiplan Plus — sei es bei der Kalkulation, grafischen Darstellung oder Dateiverwaltung — kann aufgezeichnet werden. Auf diese Weise lassen sich Folgen von Schritten zu einem Befehl zusammenfassen und auf Kommando ausführen. Diese Technik nennt man Makroverarbeitung. Bis zu 64 Makros lassen sich pro Makroblatt definieren. Die Schlüsselworte der mit 95 Funktionen sehr leistungsfähigen Makrosprache entsprechen denen des bei den PCs zur Spitzenklasse gehörenden Paketes Excel von Microsoft.

Die Aufzeichnungen selbst werden auf einem besonderen Arbeitsblatt, dem Makroblatt, festgehalten. Dort sind sie nichts anderes als eine Folge von Funktionen (Anweisungen), die geändert oder um weitere Funktionen ergänzt werden können. Natürlich können derartige Mini-Programme auch manuell erstellt werden. Zusätzliche Ein-/Ausgabe-Funktionen erleichtern die Programmierung kompletter Lösungen beispielsweise für das Verfassen von Rechnungen oder flexibler Dateiverwaltungen. Selbst an die bei Dateiverwaltungen auf dem Amiga übliche Fähigkeit, Grafiken im IFF-Format handhaben zu können, haben die Entwickler gedacht. Das Plus im Namen des Programms bedeutet übrigens »plus Makroverarbeitung«. Eine nicht programmierbare Version ist unter dem Namen »Maxiplan 500«, Preis unter 400 Mark, erhältlich.

Bei der Beurteilung des Programms führte die englische Sprache der sonst guten Dokumentation, Requester- und Menüführung zu einer Abwertung. Es soll aber eine deutsche Übersetzung bei Henrichson und Schneider in Arbeit sein. Das Prädikat absturzsicher konnten wir Maxiplan Plus leider nicht vergeben. Da diese sich in der Hauptsache durch völlige Ignoranz der Betätigung von Tasten oder Menüs äußern, sollten die Anwender der von Zeit zu Zeit ihre Arbeit sichern. Trotz dieser Mängel ist Maxiplan das zur Zeit beste Kalkulationsprogramm für den Amiga. Es beeindruckt nicht nur durch seinen Leistungsumfang, sondern auch durch programmtechnische Kleinigkeiten, die bei einer Entwicklung unter Zeitdruck wohl nicht enthalten wären. (pa)

AMIGA-WERTUNG

Software: Maxiplan Plus

9,3 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■			
Bedienung	■	■	■	■	■	■
Erlernbarkeit	■	■	■	■		
Leistung	■	■	■	■	■	

Fazit: Maxiplan Plus ist ein hervorragendes Tabellenkalkulationsprogramm. Es besitzt durch die Makrosprache eine gute Basis für die Entwicklung von Anwendungssoftware. Bisher gibt es kein leistungsfähigeres Produkt für den Amiga.

Positiv: großer Leistungsumfang; einfache Bedienung; leicht abrufbare Hilfsfunktionen; verarbeitet IFF-Dateien

Negativ: Dokumentation in Englisch; nicht absturzsicher

DATEN

Produkt: Maxiplan Plus

Preis: 798 Mark

Hersteller: Oxxi Incorporation

Anbieter: gut sortierter Fach- und Versandhandel.

Noch schneller

AMIGA test

Mit »FACC II« liegt jetzt die erweiterte Version des bekannten Amiga-Floppy-Speeders vor. Wir haben das Programm für Sie unter die Lupe genommen.

FACC und FACC II sind keine Floppy-Speeder im üblichen Sinne. Diese Art von Programmen erreicht den Geschwindigkeitsgewinn durch das Anlegen von Puffern, in denen bereits von Diskette gelesene Daten automatisch abgelegt werden. Greift der Computer nun noch einmal auf die gleichen Daten zu, erkennt dies das Programm und holt die Informationen aus dem im schnellen RAM liegenden Puffer.

So kann jetzt ein beliebig großer Speicherbereich als Puffer gewählt werden (maximal der gesamte verfügbare Speicher). Trifft der Amiga nun auf einen Befehl oder ein Programm, das er mangels freiem Speicher nicht ausführen kann, wird dies von FACC II erkannt und es werden automatisch einige seiner Pufferblöcke gelöscht, so daß wieder genügend Speicher zur Verfügung steht. Durch diese Methode und durch die Kenntnis des Amiga-DOS-Formats ist ein 500-KByte-Puffer unter FACC II zirka genauso nützlich wie ein Puffer von 1000 KByte unter dem alten FACC.

Beim Löschen der Blöcke im Speicher erkennt FACC II automatisch die »Wichtigkeit« der Blöcke und läßt zuerst die normalen Datenblöcke »fallen«. Erst wenn hier nichts mehr gelöscht werden kann, werden die Directory-Blöcke aus dem Pufferbereich gelöscht.

Um nun aber nicht immer auf eine der erwähnten »Low-Memory«-Situationen warten zu müssen, was trotz aller Maßnahmen eine gewisse Absturzgefahr bewirkt, kann vom Benutzer auch der minimale zur Verfügung stehende Fast- und Chip-RAM-Bereich vordefiniert werden. Bei der Unterschreitung dieser Grenzen werden auf die schon erwähnte Weise Puffer gelöscht.

Natürlich kann die Berücksichtigung der speziellen Blöcke beim Löschen von Pufferbereichen durch den installierten »Low-Memory-Server« oder durch direkte Puffer-Verkleinerung frei gewählt werden. So ist auch eine Arbeitsweise nach der alten FACC-Methode möglich.

Eine weitere sehr nützliche Funktion ist einfach über einen Mausklick ansprechbar. Sie ermöglicht das »Einfrieren« des Pufferinhaltes. Ist dieser Schalter angewählt, gehen keine neue Daten mehr in den Puffer von FACC II. Statt dessen werden die zu diesem Moment im Puffer befindlichen Daten eingefroren, so daß zum Beispiel einmal ein Compiler geladen (FACC II fügt ihn automatisch in den Puffer ein) und dann »Freeze« (für Einfrieren) angewählt wird. Diese Einfrierung läßt sich natürlich auch wieder rückgängig machen.

FACC II ist im Gegensatz zum FACC in zwei Teile aufgeteilt. Der Hauptteil wird einmal in den Speicher geladen und verschwindet im Hintergrund, ist also auch auf dem Workbench-Bildschirm nicht mehr im Weg. Möchten Sie nun einige Einstellungen verändern, so müssen Sie die »Oberfläche« von FACC II laden, die sich als getrenntes Programm auf der Diskette befindet (Bild).

Die zweite Möglichkeit zur Veränderung der Einstellung bietet das CLI. Damit steht einer persönlichen Konfiguration des FACC II in der Startup-Sequence nichts mehr im Weg.

Als letzte, drastische Verbesserung, auf die aber mangels Platz für diesen Testbericht nicht weiter eingegangen werden soll, ist die Möglichkeit gegeben, FACC II von eigenen Programmen aus anzusprechen beziehungsweise zu programmieren.

Speeder mit Philosophie

Nach dem nun schon wieder ein hervorragendes Produkt aus dem Hause ASDG vorliegt, soll hier einmal kurz auf die Philosophie dieses Herstellers eingegangen werden. Diese unterscheidet sich nämlich ganz entscheidend von denen anderer Firmen und verspricht einige Vorteile für alle ehrlichen Amiga-Benutzer.

ASDG ging den ungewöhnlichen Weg, sich zuerst durch Shareware einen Namen zu machen. Viele Programme wurden bereits auf den Fish-Disks veröffentlicht, und FACC war das erste kommerzielle Programm. FACC oder die resetfeste RAM-Disk »VD0:« sind mit keinem Kopierschutz versehen und bestehen nur aus wenigen Dateien. Auch die Anleitung ist nur auf Diskette vorhanden.

Aber gerade diese sehr nützlichen Eigenschaften machen solche Programme zu beliebter Raubsoftware. Darauf wird im Manual hingewiesen und um Einsicht gebeten, da die Programmierer von ASDG schon sehr viel für die Amiga-Benutzer getan haben und für ihre Mühen kaum Geld bekamen. Die Zukunft von FACC zeigte, daß sich die Besitzer dieses Programmes leider nicht sehr von diesem Vorwort angetan fühlten. FACC war bald in aller Munde und auf allen Disketten vertreten.

Nun sollten bei FACC II auch die Raubkopierer einmal zeigen, daß die »Rechtfertigung« ihres Kopierwesens wegen unverschämter Firmen und zu hohen Preisen ernst gemeint sind und waren. Diese Punkte treffen hier nämlich keinesfalls zu, und falls der Großteil der Schwarzkopierer konsequent und vernünftig ist, dürften von FACC II keine Massen-Raubkopien in Umlauf kommen. Sollte sich dieses Verfahren des Vertriebs bewähren, werden nämlich bald auch andere Firmen gute Shareware herausbringen. Ein Vorhaben, das — wie wir finden — sicherlich unterstützenswert ist.

FACC II kann allen nur ans Herz gelegt werden, die viel mit Disketten und dem Multitasking des Amiga arbeiten. Durch den »Low-Memory-Server«, der übrigens auch von anderen Programmen aus angesprochen werden kann und sollte, kann auch der Benutzer mit nur 512 KByte sehr viel mit FACC II anfangen, so daß auch dann kein Fehlkauf zu befürchten ist. Hoffen kann man hier nur, daß dieser Versuchsballon in Diskettenform nicht von den Raubkopierern abgeschossen wird. (Ottmar Röhrig/jk)

Floppy Accelerator Release II (User Interface)
Facction © 1987 by ASDG Inc.

401200 1354760	Drive 0	1	2	3	Total
Reads	1	0	585	0	586
Writes	10	0	0	0	10
ReadHits	0	0	7	0	7
Percentage	0	0	1	0	1
Buffers	5	0	0	0	1000

Fewer | Buffer Control | More
Write Retention Is On | Purge(double click) 0 1 2 3
Ignore | ChipThresh: 64 FastThresh: 512
Preference | Spc: 3 Ord: 2 Avl: 995 | U D

Die neue Benutzeroberfläche des Speeders FACC II

AMIGA-WERTUNG

Software: FACC II

10,6 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	■
Dokumentation	■	■	■	■	■	■
Bedienung	■	■	■	■	■	■
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■	■	■	

Fazit: FACC II ist ein Software-Floppyspeeder, der das Arbeiten mit Disketten, Daten und Directories auf dem Amiga beschleunigt. Er ist ein unbedingtes Muß für jeden Amiga-Besitzer, vor allem wenn ausreichend Speicher vorhanden ist.

Positiv: Programmiermöglichkeiten von anderen Programmen aus; nicht kopiergeschützt; ausführliches Handbuch auf Diskette.

Negativ: Keine Verwendung von Packalgorithmen, die eine noch höhere Speicherausnutzung ermöglichen würden; keine Unterstützung von Hard-Disks.

DATEN

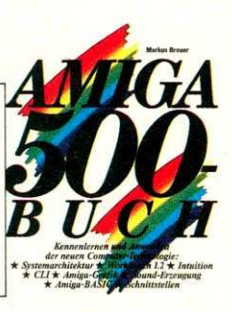
Produkt: FACC II
Preis: 49,50 Mark
Hersteller: ASDG Inc.
Anbieter: Compustore, Fritz-Reuter-Str. 6, 6000 Frankfurt, Tel. 069/567399

Die Amiga-Bibliothek

Markus Breuer
Amiga-2000-Buch
1987, 570 Seiten
Eine Pflichtlektüre für jeden, der diesen Supercomputer erfolgreich einsetzen möchte: Systemarchitektur (OSA), Umgang mit der Workbench 1.2, CLI-Befehle, MS-DOS auf dem Amiga, Umgang mit der PC-Karte, Amiga-Grafik, Überblick über Amiga-Basic.
Bestell-Nr. 90574
ISBN 3-89090-574-9
DM 59,–/sFr 54,30/öS 460,20

NEU

P. Wollschlaeger
Amiga-Assembler-Buch
1987, 329 Seiten, inkl. Diskette
Ein 68000er-Kurs mit vielen Beispielen. Mit ausführlichem Verzeichnis aller Systemroutinen und genauer Anleitung für das Einbinden von Assembler-Routinen in Amiga-BASIC.
Bestell-Nr. 90525
ISBN 3-89090-525-0
DM 59,–/sFr 54,30/öS 460,20



M. Breuer
Das Amiga 500-Buch
1987, 489 Seiten
Eine ausführliche Einführung in die Bedienung des Amiga 500. Das Handbuch dient als Nachschlagewerk beim alltäglichen Einsatz.
Bestell-Nr. 90522
ISBN 3-89090-522-6
DM 49,–/sFr 45,10/öS 382,20



F. Kremser/J. Koch
Amiga Programmierhandbuch
1987, 390 Seiten, inkl. Diskette
Eine Super-Einführung in die »Interna« des Amiga: die wichtigsten Systembibliotheken, die das Betriebssystem zur Verfügung stellt, werden ausführlich anhand von Beispielen in C erklärt.
Bestell-Nr. 90491
ISBN 3-89090-491-2
DM 69,–/sFr 63,50/öS 538,20

Dr. E. Huckert/F. Kremser
Amiga: C in Beispielen
1987, 237 Seiten, inkl. Diskette
Erste C-Programme, Daten und Datentypen, Operatoren, Befehle, Eingabe und Ausgabe, Arrays und Pinter, Strukturen und Verbunde, Prozeduren, der C-Präprozessor.
Bestell-Nr. 90539
ISBN 3-89090-539-0
DM 69,–/sFr 63,50/öS 538,20



Bantam Books
Das Amiga-DOS-Handbuch für Amiga 500, 1000 und 2000
1987, 342 Seiten
Die Pflichtlektüre für jeden Commodore-Amiga-Anwender und Programmierer: eine Entwickler-Dokumentation zum Amiga-DOS-Betriebssystem, Version 1.2. Programmierung, interne Datenstruktur und Diskettenhandling.
Bestell-Nr. 90465
ISBN 3-89090-465-3
DM 59,–/sFr 54,30/öS 460,20

Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computerfachgeschäften oder in den Fachabteilungen der Warenhäuser.

Irrtümer und Änderungen vorbehalten.

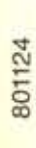


Markt&Technik
Zeitschriften · Bücher
Software · Schulung

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0.
SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656,
ÖSTERREICH: Rudolf Lechner&Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526, Ueberreuter Media Handels- und Verlagsges. mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0

COMPUTER-LITERATUR UND SOFTWARE HERBST/WINTER '87/88

Fragen Sie bei Ihrem Buchhändler nach unserem kostenlosen Gesamtverzeichnis mit über 400 aktuellen Computerbüchern und Software. Oder fordern Sie es direkt beim Verlag an!



Für Grafik nur vom Feinsten

AMIGA test

Grafik-Tools in Hülle und Fülle — Das bietet die neue Version 2.0 des hilfreichen Bildermetzgers »Butcher«. Lohnt es sich, diese stark erweiterte Version in seine Programmsammlung aufzunehmen?

Das derzeit beste Grafik-Tool, der »Butcher« (Metzger), ist in seiner neuesten Version 2.0 gegenüber dem Vorgänger V1.0 um eine Vielzahl von Funktionen erweitert worden. Außerdem liegt Butcher nun mit deutschem Handbuch und ebenfalls übersetzten Menütexten vor. Damit hebt sich dieses für Grafik-Freaks unentbehrliche Programm noch weiter in eine Position, die Butcher noch mehr zu einem der besten Programme für den Amiga macht.

Nachdem das AMIGA-Magazin bereits in Ausgabe 10/87 die Version 1.0 von Butcher getestet hat, soll an dieser Stelle hauptsächlich auf die vielen Erweiterungen eingegangen werden, die die Version 2.0 gegenüber seinem Vorgänger unterscheiden. Als erstes ist hier der neu hinzugekommene Overscan-Modus (Bilder über die gesamte Bildschirmbreite) sowie die Anpassung an die deutsche PAL-Norm (256 Zeilen) zu nennen. Dies allein schon wäre eine sinnvolle Erweiterung, gäbe es da nicht noch die anderen, erweiterten Programmpunkte.

Die bereits von der ersten Version her bekannte Funktion »Mosaik« erlaubt nun, eigene Muster zu definieren (siehe Bild). Der Anwender ist also nicht mehr nur auf die vorgegebenen, quadratischen Formen festgelegt. Die Möglichkeiten, die sich dadurch bieten, sind enorm: Grafiken können auf diese Weise einen speziellen »Anstrich« erhalten, der je nach Anwendung sicher seine Reize bietet.

Hinzugekommen ist auch der Menüpunkt »Speicherplatz«, mit dem jederzeit ein Überblick über den freien Speicher erfragt werden kann. Sollte zuwenig Hauptspeicher zur Verfügung stehen, läßt sich ähnlich wie bei Deluxe-Paint nur ein Teil des Programms laden, der Rest wird, wenn er gebraucht wird, nachgeladen. Sollte auch das nicht genügen, läßt sich von Butcher aus die Workbench abschalten sowie der UNDO-Speicher löschen, so daß weiterer Speicher gewonnen wird. Auch an Anwender, die bereits mit Deluxe-Paint arbeiten, wurde gedacht: Einige Funktionen, wie etwa Ausblenden der Menüleiste oder Starten des Color-Cycling sind wie von Deluxe-Paint her gewohnt über die gleichen Tastenkombinationen erreichbar.

Auch die Funktion »POS -> NEG«, die erlaubt, ein Negativ des Bildes zu erstellen, oder »Komplement«, mit dem jede Farbe durch ihre Komplementärfarbe ersetzt wird, sind neu hinzugekommen. Möchten Sie die Farben der Grafik in Falschfarben umwandeln? Kein Problem: »Falschfarben« erledigt dies, genauso, wie »Pseudofarben« die Farben in der Palette abhängig von der Intensität jeder Farbe regelt. Allein schon mit diesen Funktionen können beliebige Bilder so bearbeitet werden, daß ganz neue Objekte dabei herauskommen.

Brauchbar etwa für Desktop Publishing-Programme sind die Menüpunkte »Grauwert« und »Schwarz-Weiß«, die sofort aus jedem Farbbild eine Schwarzweiß-Grafik machen. Oder soll das Bild lieber einen antiken Stich erhalten? »Antik« wandelt die Farben in Brauntöne um, so daß der Eindruck eines vergilbten Fotos entsteht.

Interessant ist auch »Farbtrennen«. Damit lassen sich Bilder in einzelne Farben aufsplitten, wie sie beispielsweise bei Farbdrucken verwendet werden. Farbdrucke bestehen aus verschiedenen Farben, die man übereinander druckt. Bei vier Farbtrennungen bezeichnet man diese Farben als Schwarz, Gelb, Magenta und Cyan. Diese Funktion sorgt einfach dafür, daß jeder Auszug des Bildes der Druckfarbe möglichst nahe kommt.

Ebenfalls erweitert wurde das Zusammenfassen und Angleichen von Farben. Angenommen, es soll ein Bild so bearbeitet werden, daß es danach zwar weniger Farben besitzt, aber trotzdem nicht viel anders aussieht. Dies läßt sich mit »Konsolidierung« erzielen. Alle ähnlichen Farben werden, soweit es geht, zusammengefaßt, so daß die Farbanzahl bei fast gleicher Qualität sinkt. Den freien Platz kann der Grafiker so beispielsweise mit neuen Farben auffüllen. Das Programm wurde noch um weitere, teilweise wesentliche Punkte ergänzt. Eine Erläuterung dessen würde den Rahmen sprengen. Wir können aber behaupten, daß die Änderungen das Programm so aufwerten, daß nochmals eine Steigerung der Testnote (Version 1.0: 10,3) erzielt wurde. Und dies allein sagt schon, daß das Programm wirklich etwas Besonderes ist.

Wer Butcher 2.0 noch nicht besitzt oder noch mit der Version 1.0 arbeitet und sich für Grafik und Bildbearbeitung interessiert, sollte sich auf jeden Fall für dieses höchst leistungsfähige Grafik-Tool interessieren. Zu einem angemessenen Preis erhält der Käufer ein Programm, das nach kurzer Zeit unentbehrlich wird. Außerdem liegt dem Programm eine Aufforderung bei, Verbesserungen oder Ideen doch dem Hersteller zukommen zu lassen. Wer weiß, vielleicht können wir bald mit einer Version 3.0 rechnen...? (dm)



Mit selbstdefinierten Mustern können Grafiken bei Butcher 2.0 vielfältig verändert und manipuliert werden

AMIGA-WERTUNG

Software: Butcher 2.0 (deutsch)

10,8 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	■
Dokumentation	■	■	■	■	■	
Bedienung	■	■	■	■	■	■
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■	■	■	■

Fazit: Die gegenüber der ersten Version stark erweiterte Sammlung von Grafikroutinen bietet eine Fülle von Funktionen zur Bearbeitung von IFF-Bildern. Das einfach zu bedienende Programm unterstützt die Gestaltung und Verarbeitung von Bildern und spart dem Anwender Zeit. Dieses Programm sollte in keiner grafikorientierten Programmsammlung fehlen.

Positiv: Handbuch und Menübefehle in deutsch; sehr leistungsfähige Befehlsvielfalt; beherrscht alle Grafikmodi inklusive Overscan und PAL; umfangreiche Bearbeitungsmöglichkeiten für IFF-Bilder; nun auch Zeichenfunktion integriert

Negativ: Noch nicht genügend Möglichkeiten zur Bearbeitung von H.A.M.-Bildern; nur in Verbindung mit Zeichenprogrammen sinnvoll

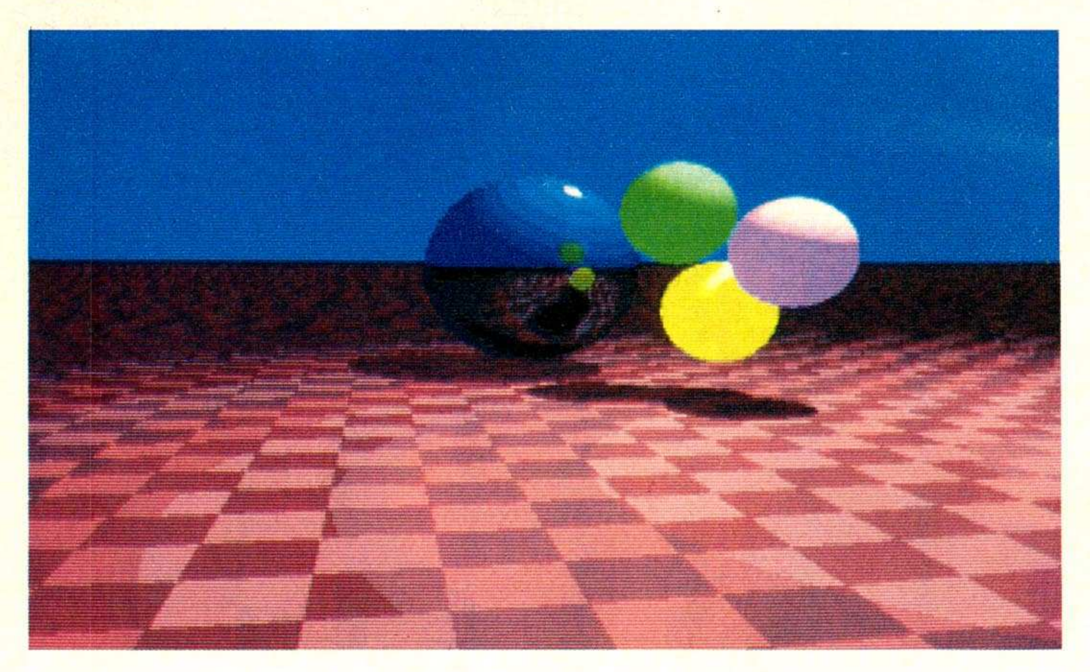
DATEN

Produkt: Butcher 2.0 (deutsch)
Hersteller: Eagle Tree Software
Preis: 79 bis 139 Mark
Anbieter: Gut sortierter Fach- und Versandhandel

Silber, aber noch kein Gold

AMIGA test

»Silver« heißt ein neues Programm aus der Ray-Tracing-Welle. Wo liegen seine Stärken und was unterscheidet es von »Sculpt 3D«?



Silver ist ein Programm zur Erstellung von Bildern nach dem Prinzip der Strahlenverfolgung. Zusätzlich können mit diesen Bildern entsprechende Filmsequenzen entworfen werden. Dazu bietet es einen normalen Ray-Tracing-Modus und außerdem einen Animationsteil, der berechnete Bilder in einer fließenden Animation zusammenfügen kann. Dies benötigt natürlich sehr viel Speicherplatz, so daß Silver nur mit mindestens einem MByte RAM läuft.

Erfreulicherweise ist festzustellen, daß Silver zu den ersten Programmen aus den USA gehört, die die deutsche PAL-Auflösung mit 640 x 512 Punkten nicht stillschweigend ignorieren. Sogar HAM-Bilder werden auf Wunsch mit Overscan und PAL-Auflösung generiert. Dabei ist allerdings nachteilig, daß keine Möglichkeit besteht, auf einem PAL-Amiga Bilder der NTSC-Auflösung zu erzeugen, was für Software-Entwickler unter Umständen von Bedeutung sein könnte.

Der größte Nachteil, selbst für den geübten Anwender, liegt bei Silver in der seltsamen Achsenbenennung. Da Silver ein dreidimensional-orientiertes Programm ist, muß natürlich auch mit drei Achsen gearbeitet werden. Die übliche Notation lautet hier X-Achse links-rechts, Y-Achse oben-unten und Z-Achse vorne-hinten.

Bilder am laufenden Band

Nicht jedoch bei Silver. Hier wird die Z-Achse zur Markierung der Höhenposition und die Y-Achse für die Abweichung vorne-hinten verwendet. Dies führt auch nach längerem Arbeiten noch zu starken Verwirrungen, was um so ärgerlicher ist, weil es programmtechnisch sicherlich kein großer Aufwand gewesen wäre, sich hier am Standard zu orientieren.

Die Erstellung der Objekte selbst ist bei Silver leider auch kein Kinderspiel, wie man es zum Beispiel von Sculpt 3D her gewöhnt ist. Um in den Editor zu gelangen, müssen Sie im Hauptmenü, das sofort nach dem Laden des Programmes angezeigt wird, die zu ändernde Szene anklicken (siehe Bild). Die Anzahl der zu belegenden Szenen ist groß genug, um auch — mit entsprechend viel Speicherplatz — schwierige Animationen zu realisieren. Jede Szene stellt bei Silver ein Bild aus der Filmsequenz dar. Die Ablauffolge der berechneten Bilder kann dann später im Animationsmenü festgelegt werden.

Sind Sie dann in den Editor für die jeweilige Szene gelangt, hört die anfängliche Begeisterung wahrscheinlich sehr schnell auf. Nicht nur, daß die Editor-Oberfläche grafisch und informativ wenig hergibt, auch als geübter Anwender läßt sich ab diesem Punkt ohne Handbuch nicht mehr viel anstellen. Objekte bekommen beim Anwählen eine andere Farbe und verschieben sich unerwartet, Quader werden zu Kugeln. Da verzweifelt ein jeder und greift zum Handbuch, von dem man sich eine Besserung dieses Zustandes erhofft.

Liegt hier vielleicht die Lösung der sich auftürmenden Probleme? Man muß diese Frage mit einem klaren »Ja« beantworten. Das Handbuch bietet eine Fülle von Informationen, die sich auf knapp 130 Seiten wie die Sardinen drängen. Das kommt durch die Kompaktheit, in der das Buch verfaßt wurde. Einem schnellen Durchlesen während der Mittagspause oder ähnlichem kann also nur abgeraten werden. Selbst wenn Sie alle Details direkt am Computer ausprobieren, werden Sie oft eine Seite zwei- oder dreimal lesen müssen, bis alles klar ist. Doch dieses Problem hat der Hersteller bedacht und eine Lösung wurde praktischerweise gleich ins Handbuch integriert.

Positionieren mit der Maus

Sie besteht hier aus viel Witz und Humor, die das Lernen erheblich auflockern und damit gleichzeitig auch intensivieren kann. Beim weiteren Lesen werden sicherlich auch Sie den Wunsch verspüren, daß mehr Handbücher auf diese Art und Weise geschrieben wären.

Um so ärgerlicher (getreu dem Spruch: Wo viel Licht ist, da ist auch viel Schatten) ist allerdings die Tatsache, daß das letzte Kapitel des Handbuches fehlt. Hier werden Tips und Tricks im Umgang mit Silver erklärt und viele wertvolle Hinweise zum Programm gegeben. Sie erhalten diese Kapitel jedoch erst nach Einsendung Ihrer Registrations-Karte. Von dieser Art des »Kopierschutzes« sollte der Hersteller schnellstens wieder absehen, da sie nicht gerade benutzerfreundlich ist. Inwieweit die deutsche Version des Handbuchs und des Programms auch noch mit dieser Art des Schutzes versehen wird, war bis zum Redaktionsschluß noch nicht abzusehen, da an der Übersetzung noch gearbeitet wird.

Das restliche Handbuch klärt übrigens auch ausführlich über die Videoaufzeichnung der gemachten Filme auf und gibt Tips zu Geräten und Schneidetechniken.

Nach getaner Arbeit — das Lesen des Handbuches artet stellenweise tatsächlich in Arbeit aus — kann dann groß eingestiegen werden. Im Editor werden mittels eines Mausklicks Objekte erzeugt, die als erstes immer im Punkt 0,0,0 stehen. Dies ist sehr problematisch, wenn Sie im Zoom-Modus außerhalb dieses Bereiches arbeiten und ein neues Objekt hinzufügen wollen. Dazu müssen Sie erst einmal wieder auf die normale Darstel-

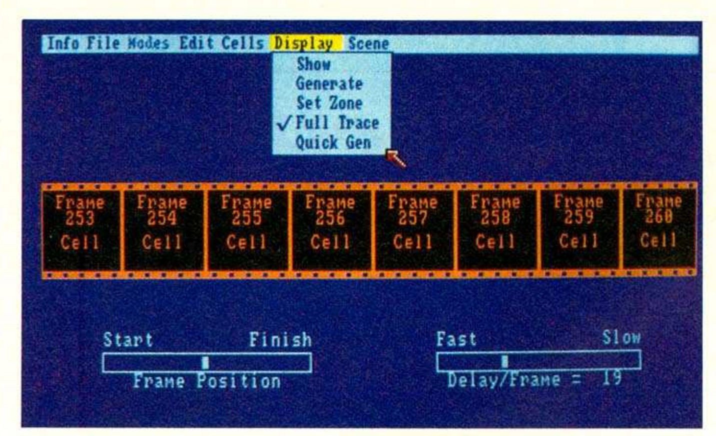


Bild. Das Hauptmenü von Silver mit den Bilder-Zellen

lungsart zurückgehen, das Objekt zeichnen und verschieben lassen und wieder zurück an die alte Stelle zoomen. Alles in allem eine zeitraubende Angelegenheit, die nicht sein müßte.

Weitere Probleme bereitet die eigenartige Darstellung der Objekte. Es werden keineswegs alle drei nötigen Blickwinkel (Oben, Vorne, Seite) gleichzeitig dargestellt, vielmehr muß zwischen diesen Blickwinkeln mittels der Maus hin- und hergeschaltet werden. Es geht auch nur mit dieser Methode, da hier keine abkürzenden Tastenkombinationen zur Verfügung stehen.

Die Koordinaten eines Objektes zum Beispiel können nur mittels der Maus, durch zweimaliges Anklicken und nachfolgendem Neupositionieren, oder mittels einiger Scrollbalken verändert werden. So ist eine genaue Koordinateneingabe oft nicht möglich, da die Auflösungen obiger Methoden zu gering sind. Gleiches Problem ergibt sich bei der zweifelsohne originell gemachten Rotation von Körpern. Hier werden unter anderem drei Windrosen angezeigt, die die Neigung des Objektes auf der jeweiligen Achse angeben. Zwar ist hier eine genaue Positionierung möglich (mit drei Scrollbalken in verschiedenen Genauigkeiten), aber diese dauert länger und ist ungewohnter als eine konventionelle Tastaturabfrage. Da die Kameraposition und deren Blickwinkel ebenfalls in dieser Art bestimmt werden, entstehen unnötige Kopfrechenaufgaben, um die gewünschten Effekte zu erzielen. Anfangs ist es sogar schwer, mittels der Windrosen überhaupt das Objekt zu »treffen«, da selbst auf dem Objektbildschirm der Kamerablickwinkel schwer erkennbar ist.

Wenn Sie Glück haben, entdecken Sie allerdings im Programm noch eine Funktion namens »Tracking«, die das Nachführen des Kamerablickwinkels auf ein bestimmtes Objekt übernimmt. Dies ist besonders nützlich, wenn Sie überflugähnliche Effekte erzielen wollen. Sie müssen dazu nur die Kameraposition ändern. Die Kamera behält das Objekt für Sie im Auge. Sind alle Einzelteile fertig editiert, können Sie diese mittels der »Bound«-Funktion verbinden. Danach werden diese Objekte nicht mehr als Einzelteile behandelt, sondern als ein Makroobjekt. Das wirkt sich zwar nachteilig auf spätere Änderungen aus, bringt aber bei der folgenden Berechnung der Szenen einen großen Zeitgewinn.

Je diffuser desto besser

Ebenfalls vor der Berechnung sollten Sie die Lichtquelle definieren. Leider kann nur ein Punkt, der sein Licht gleichmäßig in alle Richtungen abgibt, als Lichtquelle definiert werden. Hier wären mehr Optionen wünschenswert. Die Lichtquelle selbst kann in verschiedenen Farben erstrahlen und eine bestimmte Helligkeit haben. Durch Kombination mit der Diffuslicht-Funktion, die die Helligkeit des gesamten Bildes beeinflußt, sind so naturgetreue Darstellungen möglich. Weitere Verfremdungen können durch Veränderung der Kameralinse (Tele- und Weitwinkelobjektiv sind nur zwei der zahlreichen Möglichkeiten), des Horizonts beziehungsweise des Zeniths und der Behandlung von Farbübergängen erreicht werden. Natürlich kann Silver allen Objekten auch verschiedene Farben und Oberflächenbeschaffenheiten verleihen. Da diese Attribute auch auf den Untergrund zutreffen, lassen sich damit Effekte eines Sees oder Flusses naturgetreu nachahmen.

Doch nun zum Ray-Tracing-Teil von Silver. Nach der Rückkehr aus dem Editor in das Hauptmenü können Sie, ebenfalls mit der Maus, die zu generierenden Szenen anwählen. Sollten Sie auch noch die gewünschte Auflösung und ähnliche Bildschirmattribute verändern wollen, kann hier nur ein Ausstieg aus dem Programm helfen. Auflösung, Overscan und HAM-Modus können nämlich nur verändert werden, solange noch kein Objekt definiert ist. Ansonsten müssen Sie Silver neu starten, die Auflösung wählen und die Objekte wieder von Diskette laden.

Doch auch hier lauert eine Gefahr. Achten Sie nicht genau darauf, daß Sie im Objekt-Load-Requester auch eine Objektdatei angeklickt haben, kann der Amiga »abstürzen« und alle Änderungen mit sich in die unerreichbare Tiefe einer Guru-Meditation ziehen. Daß sich hier besonderer Frust einstellen kann ist nur allzuleicht verständlich. Eigentlich sollte man es schon als selbstverständlich ansehen, daß ein Programm erkennt, wenn man Konfigurationsdateien aus versehen als Objekte einladen möchte, Silver jedoch bestätigt hier wieder die berühmte Ausnahme der Regel.

Nach dem mühsamen Wechsel in die gewünschte Auflösung sollten Sie noch zwischen dem »Full-Trace«- und dem »Quick-Draw«-Modus wählen. Letzterer vernachläßigt Lichtquellen, Schatten und leider auch die Farben der Objekte, bietet dafür aber eine schnelle Übersicht über die Szene. Im Full-Trace wird das fertige Bild nach allen Regeln der Ray-Tracing-Kunst erstellt und gespeichert. Diese Speicherung erfolgt laut Philosophie der Programmierer von Silver leider keineswegs im anerkannten IFF-Standard, so daß sich eine Nachbearbeitung der Bilder als problematisch herausstellt.

Auf diese Art und Weise werden alle angewählten Bilder berechnet, was leider auch nicht sehr viel schneller als beim direkten Konkurrenten Sculpt-3D vor sich geht. Die Werbung hat hier wohl etwas zu hoch gegriffen. Eindeutige Vorteile bei der Berechnung von Bildern birgt Silver nur bei Verwendung weniger Objekte, was in der Realität wohl sehr selten vorkommt.

Die nun folgende Animation geht recht einfach vonstatten. Es werden Anfangs- und Endpunkte sowie einige weitere Merkmale der Animation angegeben und dann darf der Amiga wieder drauflos rechnen. Die einzelnen Bilder werden in einer Animationsdatei zusammengefaßt, die sich später mittels des Silver-Hauptprogrammes wieder abspielen läßt. Ein externer Recorder liegt hier leider nicht bei.

Wer hier keine Festplatte oder viel RAM besitzt, wird bald auf die Grenzen von Filmlängen stoßen — auch wenn es Ihnen nichts ausmacht, mehrere Tage auf die fertigen Bilder warten zu müssen.

Zusammenfassend muß gesagt werden, daß Silver für Animation in Sachen Ray-Tracing sicherlich zur Zeit die beste Wahl ist. Dies aber nur, weil momentan keine weiteren Programme in dieser Kategorie vorhanden sind. Sculpt-3D und der bald erscheinende Animationsteil »Animate-3D« sind dann möglicherweise die bessere Wahl.

(Ottmar Röhrig/jk)

AMIGA-WERTUNG

Software: Silver

7,5 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■	■	■	■	■
Bedienung	■	■	■			
Erlernbarkeit	■	■				
Leistung	■	■	■	■		

Fazit: »Silver« ist ein Programm, mit dem Ray-Tracing-Bilder erstellt und animiert werden können. Der Editor bietet wenig Funktionen und beschränkt sich auf eine Sicht, die nur mittels der Maus verändert werden kann. Lange Rechenzeiten machen Silver nur für Anwender interessant, die einige Tage auf ihren Computer verzichten und auf die Filme warten können.

Positiv: Multitaskingfähig; vorbildliches Handbuch.

Negativ: Verwirrende Achsenbenennung; schlechter Editor; keine IFF-Bilder; ungewohnte Rotation und Koordinatenbestimmung von Objekten; Absturz beim Laden von falschen Dateien.

DATEN

Produkt: Silver
Preis: 279 Mark
Hersteller: Impulse Inc.
Anbieter: Intelligent Memory, Basaltstr. 58, 6000 Frankfurt, Tel.: (069) 7071102

MUSIK

Fortsetzung von Seite 17

Simulierter Synthesizer

ventioneller Synthesizer simuliert werden, bei dem die Oszillator-Wellenform per Maus eingegeben wird und vielfältig moduliert werden kann. Die so erstellten Sounds können in einem speziellen Format gespeichert werden. Wie bei Deluxe-Music lassen sich Noten auch über MIDI ausgeben. Das am Bildschirm und Drucker dargestellte Notenbild läßt aber zu wünschen übrig, da keinerlei Notenbalken und Bögen vorhanden sind. Ein Außenseiter und wohl eher eine Spielerei bei Musikprogrammen ist »Instant Music« (Bild 4). Es verzichtet auf eine konventionelle Notendarstellung. Statt dessen werden die Noten durch verschiedenfarbige Quadrate angezeigt. In dieser »Schreibweise« werden komplette Songs wiedergegeben. Das Verblüffende bei Instant Music ist: Man kann ein Instrument mittels Maus selbst dazuspielen (Mouse-Jam) und dabei keinen Fehler machen. Alle unpassenden Noten und Tonlängen, die einem unterlaufen, werden automatisch der Tonleiter und dem Rhythmus der Musik angepaßt. Die Lautstärken der Instrumente lassen sich durch Schieberegler einstellen, wodurch die Helligkeit der Rechtecke zu- oder abnimmt.

Zu guter Letzt sei noch das »SoundScape Pro Midi Studio« erwähnt, das in Sachen MIDI wohl professionellste Programm. Eigentlich muß hier eher von einem Modulsystem gesprochen werden. Es gibt verschiedene Ein- und Ausgabemodule, darunter einige für MIDI und Clock. In einem Fenster kann nun ein beliebiges Eingabe-Modul mit einem beliebigen Ausgabe-Modul verbunden werden, um mit diesem Daten auszutauschen. Das Tape-Deck ist ein Modul, das alle Musikdaten aufnimmt und wiedergibt, und kann im Prinzip unbegrenzt viele Spuren aufnehmen. Mit welchem Sound eine Spur abgespielt wird und woher die Daten kommen, bestimmen einstellbare MIDI-Kanalnummern.

Der MIDI-Profi

Theoretisch kann man auf allen MIDI-Kanälen gleichzeitig aufnehmen oder 16 verschiedene MIDI-Geräte ansteuern. Die Noten werden bei SoundScape nicht als solche dargestellt, sondern als »MIDI-Events«, also eine Tabelle mit den Angaben, wann ein Ton mit welcher Tonhöhe und mit welcher Dynamik wie lange ertönen soll und vieles mehr. Da hier schlichtweg alles möglich ist und die Notendarstellung sehr undurchsichtig ist, kann SoundScape nur Musikprofis empfohlen werden. Während im Sampler-Bereich für den Amiga fast monatlich neue Produkte auf den Markt kommen, tut sich im MIDI-Bereich noch sehr wenig. Obwohl es für den Sound-Bereich bereits eine Menge guter und brauchbarer Programme gibt, muß festgestellt werden, daß noch keines dieser Programme den Amiga voll ausnutzt. Man darf auf künftige Entwicklungen gespannt sein.

(Michael Haydn/
Bernhard Carli/jk)

Fachbegriffe:

8SVX — 8 Bit Sampled Voice; Standard zur Speicherung digitaler Toninformationen auf dem Amiga

MIDI — Musical Instrument Digital Interface; Genormte serielle Schnittstelle zur Übertragung von digitalen Musikinformationen

Multisampling — Digitalisierung von mehreren Tonlagen für einen Sound

Sampler — Gerät zur Digitalisierung von analogen Tönen und Geräuschen

Sample & Hold — Digitale Abtastung eines analogen Amplitudenwertes mit darauffolgendem Konstanthalten für die Dauer der Sampling-Rate

Sampling — Vorgang der Verarbeitung analoger Sounds durch Filterung, Sample & Hold und Analog/Digital-Wandlung in digitale Informationen

Sampling-Rate — Digitale Auflösung für die Abtastung einer analogen Schwingung

Tiefpaßfilter — Filter, das tiefe Frequenzen durchläßt und hohe unterdrückt beziehungsweise abschneidet

Das sichere Doppel

AMIGA test

Sicherungskopien von Disketten sind ein lästiges, aber notwendiges Übel. Welche Hilfsmittel für einen Backup zur Verfügung stehen, erfahren Sie im Anschluß.

Bei jedem Computer ist es erforderlich, daß der Anwender immer wieder Sicherheitskopien von seinen Disketten erstellt. Gerade beim Amiga mit seinen empfindlichen Diskettenlaufwerken kommt es gelegentlich vor, daß aus verschiedenen Gründen (etwa, weil die Diskette vorzeitig aus dem Laufwerk entfernt wurde) Daten und Disketten teilweise zerstört sind. Um dem Verlust von Daten vorzubeugen, ist es ratsam, mit Sicherheitskopien zu arbeiten, damit immer ein Original zur Verfügung steht, falls doch einmal ein »Fehler« aufgetreten ist. Normalerweise lassen sich Backups (Sicherungskopien) auch mit dem Diskcopy-Befehl der Workbench erstellen. Aber erstens dauert dies relativ lang und zweitens hört das Programm zu kopieren auf, sobald ein Fehler aufgetreten ist. Im folgenden stellen wir Ihnen vier Kopierprogramme vor, die mehr können, als nur kopieren. Alle vorgestellten Programme kosten zwischen 50 und 90 Mark und sind im gutsortierten Fach- und Versandhandel erhältlich.

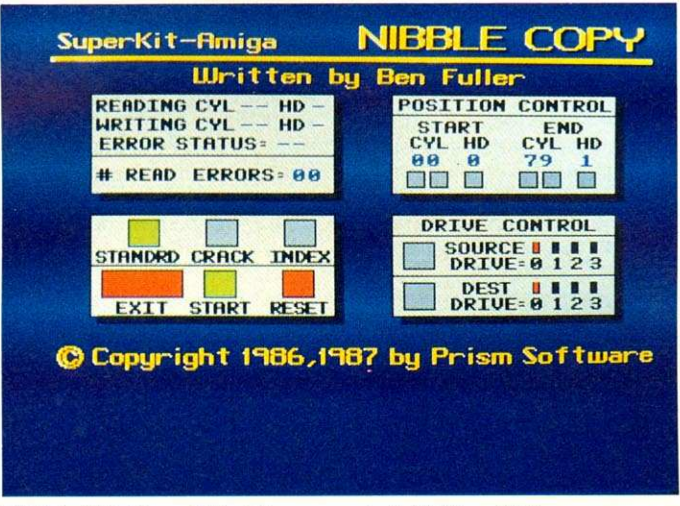


Bild 1. Mit »Superkit« können auf vielfältige Weise Sicherungskopien erstellt werden

Das erste Kopierprogramm, »Superkit« von Prism Software (Bild 1), wartet mit einer Reihe von Funktionen auf. Insgesamt besteht das Programm aus vier Modulen. Das erste ist der »Normal-Copier«. Mit ihm läßt sich eine Quelldiskette, die in einem der vier möglichen Laufwerke eingelegt sein darf, mit und ohne Verify (Kontrolle, ob die Daten richtig geschrieben wurden) auf maximal drei Diskettenstationen gleichzeitig kopieren. Selbstverständlich ist das Programm auch für Besitzer von nur einem Laufwerk verwendbar.

Die Zahl der zu kopierenden Spuren (Tracks) kann von 0 bis 79 eingestellt werden. Der Normal-Copier benötigt zum Kopieren einer normalen Diskette ohne Verify etwa 1:36 Minuten und mit Vergleichen 2:23 Minuten. Damit ist er zwar nicht der schnellste der getesteten Programme, doch liegt die Zeit gut im Schnitt.

Das zweite Modul des Superkit-Pakets ist der »Nibble-Copier«. Er kopiert auch Fehler auf der Diskette mit, so daß bei einem »Hard-Error«, an dem andere Programme scheitern, nicht gestoppt wird. Außerdem läßt sich hierbei für eine beliebige Diskette bestimmen, ob sie mit der Funktion »Standard« (Kopierzeit 1:25 Minuten), »Crack« (1:23 Minuten) oder »Index« (1:55 Minuten) kopiert werden soll. Die Anzahl der zu kopierenden Spuren kann von 0 bis 81 eingestellt werden. Wie beim Normal-Copier ist es auch hierbei möglich, auf mehrere Disketten gleichzeitig zu kopieren.

Das dritte Modul ist der »Super-Scanner«, der ähnlich wie »Diskdoctor« dazu benutzt werden kann, defekte Disketten zu behandeln und gegebenenfalls zu restaurieren. Ein weiteres Modul ist der »MFM-Editor«, der hilft, der Diskette

AMIGA-WERTUNG

Software: Superkit



7,7 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■				
Bedienung	■	■	■	■	■	
Erlernbarkeit	■	■	■	■		
Leistung	■	■	■	■		

Fazit: Von den getesteten Kopierprogrammen ist Superkit das leistungsfähigste. Wenn auch der Umgang mit dem MFM-Editor nicht ohne Grundkenntnisse erlernbar ist, bietet das Programm doch die besten Möglichkeiten.

Positiv: mit vier einzelnen Modulen leistungsfähig; im Nibble-Mode können 81 Spuren kopiert werden; bis zu vier Laufwerke werden erkannt; auch mit nur einem Laufwerk kann gearbeitet werden; eingebauter MFM-Editor; Verify wählbar

Negativ: MFM-Editor sehr einfach; Handbuch in Englisch

DATEN

Produkt: Superkit
Preis: etwa 90 Mark
Hersteller: Prism Software
Anbieter: gutsortierter Fach- und Versandhandel



Bild 2. Der »Marauder II« glänzt durch Farbenpracht und Komfort. Auch Mehrfachkopien sind möglich.

AMIGA-WERTUNG

Software: Marauder II

7,0 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■				
Bedienung	■	■	■	■	■	
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■			

Fazit: Der Marauder ist ein einfach zu bedienendes Kopierprogramm. Seine Möglichkeiten sind nicht umwerfend, genügen aber dem Anspruch an ein Kopierprogramm.

Positiv: bis zu vier Diskettenlaufwerke werden erkannt; auch mit einem Laufwerk kann kopiert werden; Verify wählbar; verschiedene Kopiermodi; eingebauter Speedcheck.

Negativ: englisches Handbuch; nur 79 Spuren

DATEN

Produkt: Marauder II
Preis: 67 bis 88 Mark
Hersteller: Discovery Software
Anbieter: Amigaland, Atlantis, Soyka; genaue Adressen entnehmen Sie bitte dem Anzeigenteil.

»aufs Byte« zu schauen. Die letzten beiden Funktionen sind eher Tools und haben mit dem Kopieren an sich wenig zu tun.

Das Superkit ist ein Kopierprogramm, das mit vielen Hindernissen fertig wird. Sollte an den Kauf eines Kopierprogramms gedacht werden, dürfte das Programm sicher in die engere Wahl fallen.

Wohl den meisten Lesern bekannt dürfte der »Marauder II« von Discovery sein. Dieses oft eingesetzte Kopierprogramm (Bild 2) gibt sich farbenfroh und leistungsstark. Der Marauder kann ebenso wie Superkit auf mehrere angeschlossene Laufwerke gleichzeitig kopieren. Aber auch mit nur einem Laufwerk können Kopien erstellt werden. Für eine durchschnittliche Diskette benötigt ein Kopiervorgang 1:24 Minuten ohne Verify und mit Vergleich 2:21 Minuten.

Falls beim Kopiervogang ein Fehler aufgetreten ist (die Arbeit wird dabei nicht unterbrochen), wird nach Abschluß der Arbeit darauf hingewiesen, daß ein Defekt registriert wurde und die Kopie nicht funktionieren könnte. Dies ist vor allem deshalb gut, da der Anwender sicher nicht die ganze Zeit auf den Bildschirm sieht, wo nur kurz eine Fehlermeldung erscheint.

Marauder kann beim Resultat, der fertigen Kopie, leider nicht immer so ganz überzeugen, obwohl zwei verschiedene Kopiermöglichkeiten, »Verbatim« und »Analytical«, zur Verfügung stehen. Auch die Funktion »Index-Sync« hilft nicht weiter, wenn »schwierige« Disketten im Laufwerk liegen. Versuche mit »defekten« Disketten ergaben, daß in allen Modi nicht immer die Kopie lauffähig war. Hier schneidet das Superkit wegen seinen größeren Möglichkeiten besser ab. Außerdem kopiert Superkit auch die Daten eines defekten Blocks mit, so daß die Chance besteht, auf der Kopie mit einem Diskettenmonitor noch etwas retten zu können.

Alles in allem ist Marauder ein brauchbares Kopierprogramm, mit dem zwar nur 79 Spuren kopiert werden können, das aber trotzdem gute Dienste bei der Vervielfältigung von Disketten leistet.

Das Copperstate-Programm »QuickNibble« (Bild 3) ist ein eher schwaches Kopierprogramm. Es kann nur mit zwei Laufwerken arbeiten und stellt auch keine großartigen Funktionen zur Verfügung. QuickNibble kann, wie die beiden anderen Programme auch, mit einem oder zwei Laufwerken betrieben werden. Es kann nur zwischen »Nibble«- und »Sector«-Copy unterschieden werden. In beiden Fällen ist die Trackgrenze bei 79 Spuren erreicht, die sich aber von 0 bis 79 variieren läßt. Das Programm ist einfach zu bedienen, da wenige Bedienungselemente angebracht sind.

Die Kopierzeiten liegen bei 1:42 Minuten für das Normal-Copy und 2:14 Minuten beim Nibble-Copy, also durchschnittliche Werte. Leider ist keine Verify-Funktion integriert, so daß erst nach verlassen des Programms festgestellt werden kann, ob nicht etwa eine defekte Diskette als Zieldiskette verwendet wurde.

Sollten keine zu großen Ansprüche an ein Kopierprogramm gestellt werden, kann das Programm sicher Verwendung finden. Es sollte aber überlegt werden, ob man nicht lieber doch etwas mehr Geld in ein leistungsfähigeres Programm investieren will.

Wohl nur für Besitzer von nur einem Diskettenlaufwerk interessant ist »White Lightning« von Vesaliasoft (Bild 4). Das Kopierprogramm besitzt eine zuschaltbare Verify-Funktion, wodurch Kopierfehler vermindert werden. Ohne Verify wird (Diskettenwechsel eingerechnet) eine normale Diskette in 1:41 Minuten dupliziert, das Kopieren mit Vergleich benötigt 2:20 Minuten. Vorteilhaft ist, daß angegeben werden kann, ob bei einem Lesefehler weitergemacht werden soll.

White Lightning offenbart seine Stärken erst dann, wenn ein Amiga mit mindestens 1 MByte Speicher bereitsteht. In diesem Fall kann der komplette Disketteninhalt auf einmal im Speicher gehalten werden. Wenn nun von einer Diskette mehrere Kopien anzufertigen sind, muß dadurch nicht jedesmal die Quelldiskette wieder eingelegt werden, sondern die Zieldiskette wird direkt aus dem Speicher heraus beschrieben, das Kopieren geht also schneller, je mehr Zieldisketten man beschreibt.

Zu guter Letzt sei noch erwähnt, daß das Pogramm maximal 81 Spuren kopieren kann. Die Gesamtanzahl läßt sich von 0 bis 81 einstellen.

Für welches der vorgestellten Programme Sie sich auch entscheiden werden, es ist immer ein besseres, sichereres und komfortableres Arbeiten als mit dem Workbench-Befehl »DiskCopy«. (dm)



AMIGA-WERTUNG

Software: White Lightning

5,8 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	x	x	x			
Dokumentation	x	x				
Bedienung	x	x	x	x		
Erlernbarkeit	x	x	x	x	x	
Leistung	x	x				

Fazit: Für Besitzer von nur einem Laufwerk ist das Programm eine überlegenswerte Alternative. Durch seinen geringen Preis und die angemessene Leistung sollte White Lightning sicher in die engere Wahl fallen.

Positiv: Mehrfachkopien möglich; Verify wählbar; bis 81 Spuren; Speicheranzeige; bei genügend Speicher Mehrfachkopien einfach.

Negativ: arbeitet nur mit einem Laufwerk; sinnvoll erst ab 1 MByte Hauptspeicher.

DATEN

Produkt: White Lightning
Preis: 49 Mark
Hersteller: Vesaliasoft
Anbieter: Vesalia Versand, Marienweg 40, 4230 Wesel, Tel. (0281) 65466 - 62205

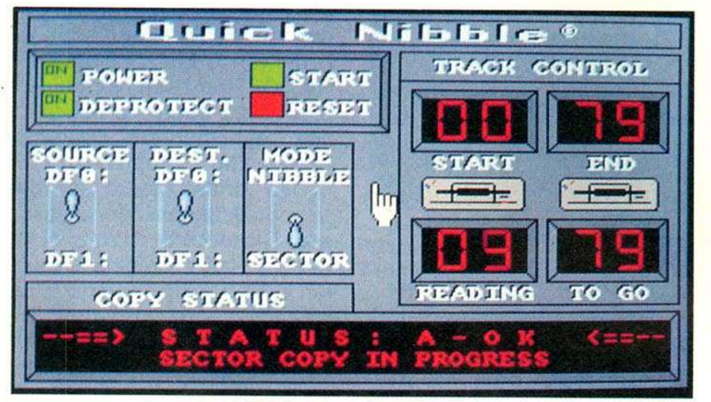


Bild 3. »QuickNibble« ist eher ein einfaches Nibble-Programm mit wenig Komfort

AMIGA-WERTUNG

Software: QuickNibble

5,3 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	x	x	x			
Dokumentation	x	x				
Bedienung	x	x	x			
Erlernbarkeit	x	x	x			
Leistung	x	x				

Fazit: QuickNibble ist ein Nibble-Programm mit geringen Möglichkeiten. Es arbeitet nur mit zwei Laufwerken und stellt auch keine Verify-Funktion zur Verfügung.

Positiv: einfach zu bedienen; Nibble- und Sectorcopy; kann auch mit nur einem Laufwerk betrieben werden.

Negativ: kein Verify; kann nicht mehr als 79 Spuren kopieren; nur zwei Laufwerke werden erkannt.

DATEN

Produkt: QuickNibble
Preis: etwa 79 Mark
Hersteller: Copperstate
Anbieter: Amigaland, Hohenwaldstr. 26, 6374 Steinbach, Tel. (06171) 71846

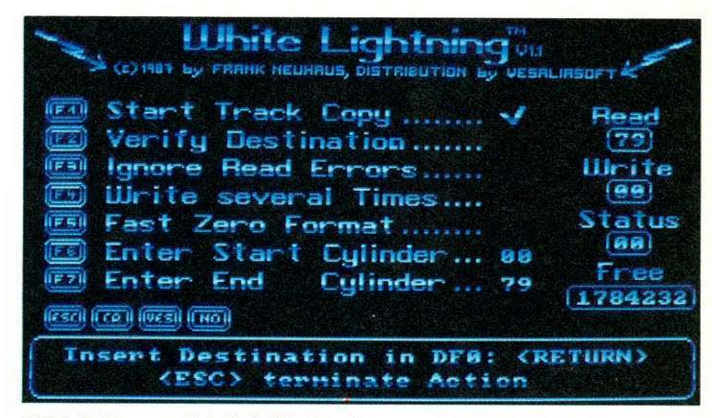
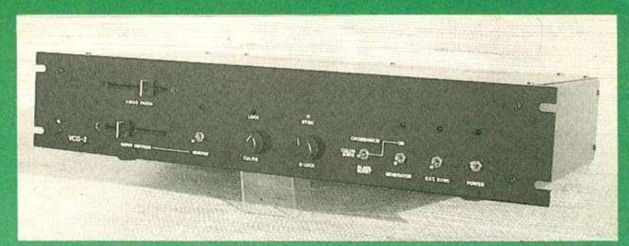


Bild 4. Hauptsächlich für Besitzer von nur einem Laufwerk gedacht: »White Lightning«

AMIGA VIDEOACTION PROFESSIONAL



HV-1 Home-Video Genlock

Passend für Amiga 500/1000/2000. Hardwaresteuerung und Auto-Configuration. Speziell geeignet für VHS und Beta sowie alle anderen Video-Systeme. Verlustfreie Signalverarbeitung.

HV-1 DM 1198,-

VCG-2

Broadcast Genlock

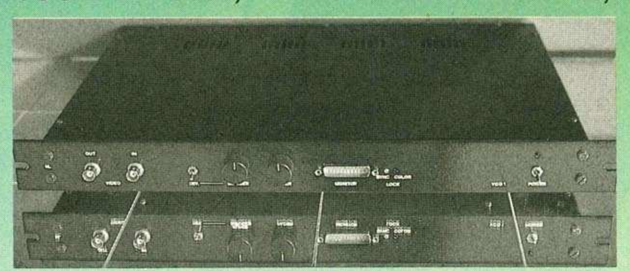
Ein absolut professionelles Gerät für alle Amiga-Typen. Vielfältige Einstell- und Anschlußmöglichkeiten. Exzellente Bildqualität. Für alle Videosysteme geeignet, besonders leistungsfähig mit 1″-, U-matic-HIGH/LOWBAND- u. BetaCam-Recordern.

VCG-2 DM 3998,- VCG-2 RGB DM 4498,-

PalVideo 2000 RGB-Coder

Macht aus Ihrem Amiga 2000 einen vollwertigen PAL-Amiga. Getrennte Ausgänge für Monitor und BNC-Video. Steckfertiges Modul.

PALVideo 2000	DM 198,-
Video 500 für A500	DM 128,-



VCG-1 Studio Genlock

Genlock-Interface für gehobene Studio-Ansprüche zum Einsatz im U-matic-HIGH/LOWBAND-Schnittplatz sowie f. SuperBeta, VHS, Video 8 oder andere Systeme. VCG-1 DM 1998,-

megamiga® RAM-Erweiterung

1-MByte-Speicherkapazität für den Amiga 1000. Der Einbau erfolgt »unsichtbar« hinter der Frontabdeckung. Kompatibel durch mitgelieferte MegaKick®-Disk. Die Karte wird für die untenstehenden Programme empfohlen bzw. benötigt.

MegAmiga mit MegaKick DM 498,-

Professional Equipment:

GSE VEC-1	Schnittsteuersystem für VHS, Beta und Video 8	**DM 1998,-**
GSE VTC-1000	Timecode-Schnittsteuersystem U-matic/VHS/Beta	**DM 3798,-**
VCP-1	RGB-Farbprozessor mit Effektmögl. und Verst.	**DM 1198,-**
Panasonic Videokamera F-10		ab **DM 2398,-**
SONY KV-1440 Black-Trinitron-Monitor/TV für Amiga		**DM 798,-**
Nachleuchtender Hires-Monitor, flimmerfreier Interlace-Mode		**DM 3498,-**

Weitere Videokameras, Videorecorder, Monitore, Videoeffektgeräte und Videomischer auf Anfrage.

Video-Software:

Grafik

De-Luxe Paint II Deutsche PAL-Version	**DM 249,-**
Digi-Paint Deutsche PAL-Version	**DM 139,-**
Master CAD 3D	**DM 168,-**
Butcher 2.0 PAL	**DM 89,-**
Sculpt 3D	**DM 198,-**

Animation

Videoscape 3D	**DM 389,-**
Silver 3D Ray Tracing Animator	**DM 289,-**
Forms in Flight 3D CAD	**DM 169,-**
Disney 3D Animator	**DM 589,-**
Disney 3D Junior	**DM 198,-**

Titel

JDK ProVideo CG-1 Video Titler, 100 Seiten, PAL-Version	**DM 398,-**
Deutscher Zeichensatz in Vorbereitung	**DM 98,-**

VIDEOCOMP
Video + Computer
Dipl.-Kfm. K. J. Leuze

MICHAEL LAMM
COMPUTERSYSTEME

Professional Video Beratung & Vertrieb
VideoComp
Bergerstr. 193
6000 Frankfurt/M. 1
Tel.: 069/467001

Computerzubehör Beratung & Vertrieb
LAMM Computersysteme
Schönbornring 14
6078 Neu-Isenburg 2
Tel.: 06102/52535



Hardware und Software
Herstellung-Vertrieb
Professional Video
Elektron. Bauteile

Alle Preise verstehen sich zuzüglich Porto und Verpackung. Preisänderungen vorbehalten.

SOFTWARE FÜR



13 Superprogramme für den Amiga:
Grafik • stop • Musik • stop • Video • stop • Datenbank • stop • Entwicklungspaket • stop • Und viele weitere Top-Hits...



7

AMIGA EXTRA
Nr. 1
GRAFIK
CADos 3D
Konstruktion und Rotation dreidimensionaler Körper
Fractal Construction Kit
Bilder einer fremden Welt
Funktionsplotter
Grafische Auswertung komplexer Funktionen
VOLUME 1

8

SOFTW
AMIGA
Markt&Tec

6

DELUXE
VIDEO
ELECTRONIC ARTS

1

Deluxe Paint II (deutsch)
Dieses Grafikprogramm ist eines der außergewöhnlichsten auf dem Softwaremarkt. Mit ihm sind Sie in der Lage, in der Textfunktion Fett- und Kursivschrift sowie Unterstreichungen einzubauen, perfekte Manipulation in der dritten Dimension darzustellen oder wichtige Parameter wie Längsdruck, Schwarzweißgraustufen- oder Farbdruck, Ränder und vieles mehr unmittelbar vor dem Druck per Mausklick zu verändern.
Hardware-Anforderung:
Amiga (512 Kbyte), Farbmonitor.
Bestell-Nr. 52584
DM 249,–*
(sFr 225,–*/öS 2990,–*)

2/3

Die ideale Ergänzung zu den Deluxe-Programmen Paint, Print und Video: Seasons & Holidays und Art Parts II. Diese Applikationen beinhalten eine Sammlung von Bildern, die in eigene Bildkreationen eingegliedert werden können.
Deluxe Seasons & Holidays
Bestell-Nr. 52580
DM 29,–* (sFr 24,–*/öS 290,–*)
Deluxe Art Parts II
Bestell-Nr. 52581
DM 29,–* (sFr 24,–*/öS 290,–*)

4

Deluxe Music (deutsch)
Das professionelle Musikprogramm, das den Ton angibt! Bei der Wiedergabe, beim Publizieren und beim Komponieren. Mit Deluxe Music sind Sie Komponist und Dirigent: Noten erst sehen und hören, dann plazieren, Noten und Taktstriche auswählen und verschieben, komplette Schneide-, Klebe- und Kopierfunktionen, Schrittmodus-Eingabe über das MIDI-Keyboard oder die Bildschirm-Tastatur und noch vieles mehr.
Hardware-Anforderung:
Amiga (512 Kbyte).
Bestell-Nr. 52579
DM 199,–*
(sFr 179,–*/öS 2490,–*)

5

Deluxe Print I (englisch)
Dieses Programm und ein grafikfähiger Drucker verwandeln den Amiga in eine Druckmaschine. Sie können Karten, Poster, Briefköpfe und vieles mehr auf einfachste Weise entwerfen und ausdrucken. Als Besitzer eines Farbdruckers können Sie Ihr Werk auch in Farbe aufs Papier bringen. Deluxe Print ist kompatibel zu Deluxe Paint. Sie können also auch Grafiken zwischen den Programmen austauschen!
Hardware-Anforderung:
Amiga (512 Kbyte), Farbmonitor.
Bestell-Nr. 52566
DM 99,–*
(sFr 89,–*/öS 990,–*)

6

Deluxe Video 1.2 (englisch)
Mit Deluxe Video können Sie animierte Grafik-Sequenzen einfach entwerfen und zusammenstellen. Videofilme lassen sich mit Computergrafik versehen, und Sie können regelrechte Computer-Videoclips zusammenstellen. Das Programm ist außerdem kompatibel zu Deluxe Paint und Deluxe Video. Sie können also auch Grafiken zwischen den Programmen austauschen.
Hardware-Anforderung:
Amiga (512 Kbyte), Farbmonitor.
Bestell-Nr. 52567
DM 249,–*
(sFr 225,–*/öS 2990,–*)

* Unverbindliche Preisempfehlung

801275



7
Amiga Extra Grafik, Vol. I
Diese Programmsammlung nutzt die außergewöhnlichen Grafikfähigkeiten Ihres Amiga 500, 1000 und 2000: CADos 3D ermöglicht Ihnen u.a. das Konstruieren komplexer dreidimensionaler Körper, mit Fractal Construction Kit können Sie mathematische Erkenntnisse grafisch umsetzen und das Programm Funktionsplotter bietet Ihnen u.a. die maßstäbliche Darstellung von Funktionen und den Vergleich mehrerer Funktionen.
Bestell-Nr. 38708
DM 49,–*
(sFr 45,–*/öS 599,–*)

8
Zing! – das mächtige CLI-Werkzeug (deutsch)
Mit Zing! haben Sie endlich das gesamte File-System mit Directories und Subdirectories fest im Griff. Die Bedieneroberfläche ist vom Feinsten: Pull-down-Menüs, (Click-)Icons, Funktionstasten und noch viele weitere Optionen.
Hardware-Anforderung:
Amiga 500, 1000, 2000
mit 512 Kbyte RAM, Kickstart 1.2.
Bestell-Nr. 51669
DM 189,–*
(sFr 169,–*/öS 2290,–*)

9
Zing Keys! – Ihr ganz persönlicher Amiga (deutsch)
Zing Keys! bietet jetzt ein formbares Software-Werkzeug mit erstaunlichen Fähigkeiten. Mit Zing! Keys! machen Sie aus Ihrem Amiga das variable System, das Sie sich schon immer wünschen. Alle Tasten sind nach Wunsch belegbar, und die Belegung ist natürlich jederzeit abspeicherbar. Durch die Belegung von Hot-Keys haben Sie mit Zing Keys! ein Multitaskingsystem par excellence!
Hardware-Anforderung:
Amiga 500, 1000, 2000
mit 512 Kbyte RAM, Kickstart 1.2.
Bestell-Nr. 51670
DM 99,–*
(sFr 89,–*/öS 990,–*)

10
CLImate 1.2 (deutsch)
Mit CLImate 1.2 können Sie endlich die Befehle des Command-Line-Interface benutzerfreundlich per Mausklick verwenden! Eine sehr übersichtliche Bildschirmdarstellung, die Bedienung aller Befehle mit der Maus und die Unterstützung von drei externen Laufwerken (3½" oder 5¼"), Festplatten, RAM-Disk machen das Programm zu einem unentbehrlichen Werkzeug.
Hardware-Anforderung:
Amiga 500, 1000 oder 2000 mit mindestens 512 Kbyte Hauptspeicher. Empfohlene Hardware: Farbmonitor. Software-Anforderung: Kickstart 1.2 (oder ROM bei Amiga 500 und 2000), Workbench 1.2.
Bestell-Nr. 51653
DM 79,–*
(sFr 72,–*/öS 990,–*)

11
Superbase Amiga
Superbase Amiga vereint eine neuartige, äußerst benutzerfreundliche Bedienung mit Pull-down-Menüs, Fenstern und Maussteuerung mit der enormen Leistungsfähigkeit einer relationalen Dateiverwaltung: Sie können Datenbanken einfach und schnell aufbauen, Daten übersichtlich verwalten, z.B. als Tabelle oder Formular, beliebige Datenkategorien auswählen, Dateien verknüpfen und einbinden, Bilder und Grafiken darstellen, einzigartige Dia-Shows erstellen oder Grafik-Datenbanken verwalten.
Bestell-Nr. 51636
DM 249,–*
(sFr 199,–*/öS 2990,–*)

12
VideoScape 3D – ein neues Programm für die Amiga
Mit VideoScape 3D können Sie dreidimensionale Objekte aus verschiedenen Blickwinkeln ansehen und durch Hinzufügen von Kamerafahrten und frei wählbarem Lichteinfall einen realistischen Computer-Videofilm erstellen. Das Programm ist an die PAL-Auflösung angepaßt und wird mit deutschem Handbuch ausgeliefert.
Bestell-Nr. 51671
DM 385,–*
(sFr 345,–*/öS 3850,–*)

13
Devpac Assembler (deutsch)
Ein Entwicklungspaket mit integriertem Editor/Assembler, symbolischem Debugger und schnellem Linker zum Einbinden von Hochsprachen-Modulen. Erzeugt direkt ausführbare Programme!
Bestell-Nr. 51656
DM 148,–*
(sFr 134,–*/öS 1480,–*)

* Unverbindliche Preisempfehlung.

Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computer-Fachgeschäften oder in den Fachabteilungen der Warenhäuser.

Markt&Technik
Zeitschriften · Bücher
Software · Schulung

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0
Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656
ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526
Ueberreuter Media Verlagsges. mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0.

Emulieren müßte man können

AMIGA test

Wer vom C 64 auf den Amiga umgestiegen ist, kennt das Problem: Die reichlich vorhandene C 64-Software läuft nicht auf dem neuen Computer. Da liegt doch die Idee nahe, einen Emulator zu schreiben: »GO 64!« und »64-Emulator« sollen den C 64 in den Amiga zaubern. Geht das?

Um überhaupt von C 64-Disketten Programme einlesen zu können, werden der Amiga und die zum C 64 gehörende Floppy 1541 mit einem Anschlußkabel (64-Emulator, Bild 1 rechts) oder mit einer Anschlußbox (GO 64!, Bild 1 links) ausgeliefert. Durch diese Verbindung über den seriellen Port des Commodore 64 können alle C 64-Peripheriegeräte (in erster Linie natürlich die Diskettenlaufwerke) an den Amiga angeschlossen werden. Dieser Anschluß erfolgt am Amiga bei beiden Emulatoren über den parallelen Druckerport, so daß ein vorhandener Drucker für die Zeit der C 64-Emulation ausgesteckt werden muß.

Soll die Emulation gestartet werden, so muß man einen Reset des Computers durchführen und die 64-Emulator-Disk einlegen. GO 64! wird von der Workbench aus gestartet. Leider sind beide Programme kopiergeschützt, was besonders beim GO 64! nicht sehr sinnvoll ist, da er ohne das C 64-Interface sowieso nicht verwendet werden kann. Ist beim Amiga 500 eine Speichererweiterung aktiviert, so funktioniert die Erkennung der Originaldiskette durch den GO 64! nur bei etwa 50 Prozent der Ladeversuche.

Wurde der 64-Emulator geladen, so fällt sofort auf, daß die Programmautoren auf den linken und rechten Rand des Bildschirms (laut Anleitung aus Zeitgründen) verzichtet haben. GO 64! dagegen stellt diese Ränder dar. Die Tastaturabfrage und das Scrolling des 64-Emulator sind allerdings auch deutlich schneller als die des GO 64!. Bei letzterem muß man oft eine Taste noch einmal drücken, da die Abfrage nicht schnell genug ist. Das Scrolling erfolgt hier ruckartig in Blöcken von fünf Zeilen, wodurch besonders das Lesen von Listings nicht gerade erleichtert wird. Beim 64-Emulator ist das Scrolling fließend, allerdings fehlt hier eine Taste, um die Ausgabe zu verlangsamen, da die CTRL-Taste nicht belegt wurde. Alles in allem vermittelt der 64-Emulator einen deutlich besseren Eindruck als der GO 64!.



Bild 1. Beide Commodore 64-Emulatoren mit ihrer Hardware; 64 Emulator (rechts), GO 64! (links)

In der Tastenbelegung unterscheiden sich die beiden Emulatoren bei den Sonder- und Steuertasten erheblich. Offenbar hat man beim GO 64! versucht, die Position der Sonderzeichen den Positionen auf der C 64-Tastatur so ähnlich wie möglich zu machen. Die Tastenbelegung des 64-Emulator ist dagegen sehr stark an die amerikanische Amiga-Tastatur angelehnt, was allerdings auf einem deutschen Amiga zum Umdenken zwingt. Die Zehnertastatur des Amiga ist bei beiden Emulatoren im C 64-Modus aktiv.

Unterschiedliche Tastenbelegung

Ist GO 64! geladen, so fällt unangenehm auf, daß bei eingesteckter Maus überhaupt nichts getippt werden kann und nur wirre Zeichen auf dem Bildschirm erscheinen. Praktisch ist dagegen, daß bei diesem Emulator die Standard-Geräteadresse der Floppy auf 8 umgeschaltet wurde, so daß das lästige »,8« nach dem LOAD-Befehl entfällt. Das Phänomen der blockierten Tastatur tritt übrigens auch beim Original-C 64 auf, wenn ein in Port 1 gesteckter Joystick bewegt wird. Hier geht der 64-Emulator einen wesentlich besseren Weg: Für jeden Controlport wird mit Hilfe eines Kontrollfensters (Bild 2), das mit <CTRL HELP> aufgerufen wird, ein zu simulierendes Gerät (Joystick, Paddle oder Maus) ausgewählt. Ist der Port 1 auf »Paddle« geschaltet, so kann die Maus eingesteckt bleiben. Bei vielen Spielen muß man diesen Port allerdings auf »Joystick« umschalten. Durch eine Maus-Simulation können sogar einige neuere Programme, die mit den C 64-Mäusen 1350 oder 1351 arbeiten, mit der Amiga-Maus benutzt werden.

Mit dem Kontrollfenster (Bild 2), das sich auf Tastendruck »über« das momentan benutzte Programm legt, werden auch alle anderen Systemparameter des 64-Emulator eingestellt. Für grafikintensive Programme ist etwa die Umschaltung auf monochrome Darstellung nützlich, da so die Ablaufgeschwindigkeit erhöht wird. In einem Amiga mit 1 MByte Speicher läßt sich sogar die 1750-Speichererweiterung von Commodore simulieren. Das Kontrollfenster hat nur einen Nachteil: Da alle angezeigten Parameter nur mit der Maus geändert werden können, muß ständig umgesteckt werden, wenn ein Joystick in Port 1 benötigt wird. Hier sollte eine Steuerung des Mauszeigers mit den Cursortasten eingebaut werden. Der GO 64! geht hier einen anderen Weg: Um seine Parameter (etwa die Abfrage der Paddles) einzustellen, muß man den Amiga neu starten und ein eigenes Einstellprogramm aufrufen.

Natürlich ist es beim Kauf eines C 64-Emulators besonders wichtig, daß möglichst viele Programme auch mit ihm funktionieren. Spiele bereiten in dieser Beziehung die größten Probleme: Obwohl die beiden Emulatoren auch komplizierte Grafikprogrammierungen des C 64 verstehen können, laufen die meisten Originalspiele wegen des Kopierschutzes nicht. Hat man ein Spiel doch zum Laufen gebracht, ist man von der sehr geringen Ablaufgeschwindigkeit enttäuscht. Außerdem wird die komplizierte Musik vieler neuerer Spiele vom Amiga nur sehr schlecht wiedergegeben, wobei der 64 Emulator deutlich besser als der GO 64! abschneidet. Der 64-Emulator stellt die Sprites des C 64 zwar dar, sie flimmern aber oft sehr stark. Das von manchen Programmen verwendete Soft-Scrolling wird von beiden Emulatoren nur sel-

Bild 2. Das Kontrollfenster des »64-Emulator«

ten richtig ausgeführt. Insgesamt sind beide Emulatoren für Spiele schlecht geeignet.

Viel besser sieht es bei Anwenderprogrammen aus, bei denen wohl auch der Hauptanwendungsbereich der Emulatoren liegen dürfte. Da die Grafik bei diesen Programmen nicht so intensiv benötigt wird, laufen sie oft nur wenig langsamer als auf dem C 64. »Superbase« oder »Vizawrite« etwa lassen sich mit dem 64-Emulator durchaus in einer akzeptablen Geschwindigkeit verwenden, obwohl die Umlaute bei Vizawrite nicht an der richtigen Stelle auf der Tastatur liegen. Bei der Kompatibilität hat der 64-Emulator deutlich die Nase vorn — Vizawrite zum Beispiel ist mit GO 64! nicht lauffähig.

Kopierschutz-Probleme

Die höhere Kompatibilität des 64-Emulator ist unter anderem darauf zurückzuführen, daß er Kopien der ROM-Bausteine des C 64 enthält, während diese beim GO 64! lediglich simuliert werden. In der Anleitung dieses Emulators wird zwar beschrieben, wie die C 64-ROMs auf den Amiga überspielt werden können; dazu werden aber wahrscheinlich nur die wenigsten Benutzer imstande sein. Auch bei den Anwenderprogrammen gilt, daß Versionen ohne Kopierschutz eher funktionieren als Versionen mit Schutz.

Am besten funktionieren die beiden Emulatoren mit Basic-Programmen, da diese meist nicht so intensiv mit den Hardware-Bausteinen des C 64 arbeiten und von der Geschwindigkeit her nicht so kritisch sind. Die Ausführungsgeschwindigkeit der Programme ist insgesamt beim 64-Emulator deutlich höher als beim GO 64!. Das Laden vom 1541-Laufwerk dauert allerdings bei beiden Emulatoren ebenso lang wie beim Original. Floppy-Beschleuniger wie Hypra- oder Ultraload lassen sich bei keinem der beiden Programme verwenden.

Um die Ladezeiten zu verkürzen, geht der 64-Emulator einen anderen Weg: Oft benötigte Programme werden mit Hilfe eines Hilfsprogrammes auf Amiga-Disketten oder Festplatte umkopiert und dann mit der vier- bis fünffachen Geschwindigkeit des C 64-Laufwerkes geladen. Dabei werden die Programme im normalen Amiga-DOS-Format gespeichert. Das Umkopieren dauert allerdings relativ lange. Obwohl der C 64-Emulator 1541-Floppykommandos zum Löschen von Dateien akzeptiert, werden sie auf Amiga-Disketten nicht ausgeführt. Um eine Datei zu löschen, muß also der Emulator verlassen werden.

Wenn nötig, kann auf einer Amiga-Diskette sogar ein 1541-Laufwerk simuliert werden, das die normalen C 64-Floppy-Kommandos für das direkte Bearbeiten von Blöcken und die Verwendung von relativen Dateien versteht. Das Übertragungsprogramm kann auch eine ganze 1541-Diskette mit allen Blöcken in die simulierte 1541 auf dem Amiga-Laufwerk kopieren (was aber etwa 20 Minuten dauert). Somit ist es ohne weiteres möglich, das Dateiverwaltungsprogramm Superbase — das ja direkt auf die Diskette zugreift — mit einem Amiga-Laufwerk zu verwenden. Wird ein Diskettenmonitor geladen, kann die simulierte

Fortsetzung auf Seite 138

Haben Sie gewonnen?

Die Sieger des Malwettbewerbs stehen fest! Die schönsten Bilder des Grafikwettbewerbs aus Ausgabe 6/7 wurden durch Ihre Mithilfe ermittelt. Lesen Sie, wer die besten sind und ob auch Sie einen unserer als Mitmach-Dank ausgesetzten Preise gewonnen haben.



Bild 1. Der Fantasy-Rider von Jens Eisert erreichte den 1. Platz

Es ist vollbracht! Die schönsten Bilder des Grafikwettbewerbs sind ermittelt. Durch Ihre Mithilfe bei der in Ausgabe 10 ausgeschriebenen Entscheidung konnten klar die Sieger bestimmt werden.

Von den insgesamt 568 Postkarten, die uns erreichten, fielen allein 102 Stimmen auf den Fantasy Rider von Jens Eisert (Bild 1). Sein Bild belegt den ersten Platz. Wir gratulieren ihm zu seinem Preisgewinn von 600 Mark und hoffen, daß noch viele wunderbare Grafiken von ihm entstehen werden.

Platz zwei fiel mit 76 Stimmen auf Futuretown von Rainer Reber (Bild 2). Auch er kann den verdienten Gewinn von 250 Mark sein eigen nennen. Vielleicht erreicht uns in Zukunft noch das eine oder andere Bild von ihm?

Das Bild, das mit 63 Stimmen auf dem dritten Platz landete, stammt von G. Holzheimer. Seine Galleone (Bild 3) brachte ihm 150 Mark ein.

Übrigens sind alle Bilder, die an der Endausscheidung teilgenommen haben, auf unseren IFF-Bilderdisketten, die Sie für 29,90 Mark (2 Disketten) beim Programmservice (Bestell.-Nr. 49705) bestellen können, enthalten.

Doch wie bereits erwähnt, haben nicht nur die Zeichner der Bilder, sondern auch einige der Leser, die uns bei der Wahl der Besten geholfen haben, etwas gewonnen. Die insgesamt 15 Preise (drei Jahresabonnements des AMIGA-Magazins und 12 Bilderdisketten) wurden aus dem großen Haufen der Postkarten gezogen. Je ein Jahresabonnement erhalten:

Marcus Hammermeister, 3043 Schneverdingen
Robin Kositza, 3180 Wolfsburg
Elke Lösch, 6094 Bischofsheim

Die weiteren Preise, jeweils zwei Disketten mit den Wettbewerbsbildern, gehen an:

Jens Wolfram, 3307 Schöppenstedt
Martin Mayer, 8012 Ottobrunn
Mark Norbert, A-1220 Wien
Gudrun Färber, 8170 Bad Tölz
Michael Berger, 4060 Viessen
Wolfgang Kierdorf, 5060 Leverkusen
Holger Koschinski, 4000 Düsseldorf
Sarah Vrigt, 1000 Berlin
Markus Pfeiffer, 3507 Baunatal
Phillip Schaper, 2000 Hamburg
Ralf Eichler, 7000 Stuttgart
Sven Lindner, 3016 Seelze

Wir bedanken uns bei allen Einsendern, die uns Postkarten zur Gewinnermittlung schickten sowie natürlich bei allen Grafikern. (dm)



Bild 2. Unser zweiter Sieger: Futuretown von Rainer Reber



Bild 3. Die Galeone von G. Holzheimer, der dritte Sieger

Die Werkzeugkiste für Anwender

Um seinen Computer zu beherrschen, sind Tools und Utilities (Hilfsprogramme) eine unabdingbare Voraussetzung. Sie erleichtern den Umgang mit dem Computer und vereinfachen das Arbeiten.



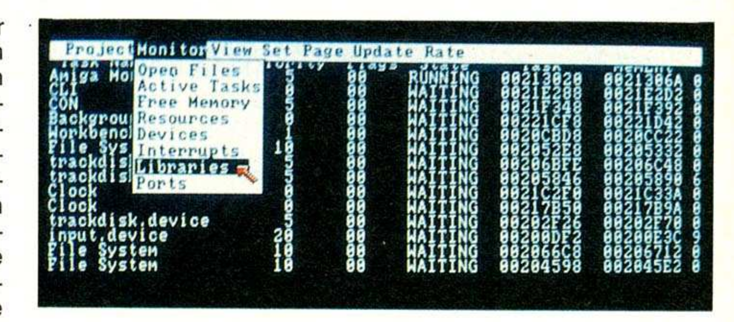
Im Umgang mit dem Amiga treten immer wieder umständliche Bedienungsvorgänge auf, die bestimmt auch einfacher zu lösen wären. Nicht immer müssen lange Tastaturkommandos eingegeben werden, denn wozu besitzt unser Computer eine grafisch orientierte Benutzeroberfläche mit einem funktionierenden Multitasking-Betriebssystem? Was liegt näher, als dem Anwender die Tastatureingaben abzunehmen und Aktionen durch Mausbenutzung zu vereinfachen? Beispielsweise könnte dafür das Programm »CLI-Mate« genannt werden, mit welchem sich Dateioperationen wesentlich vereinfachen, da alle Funktionen mit der Maus aktiviert werden.

Oder sei es nur, daß der Benutzer durch Hilfsprogramme einfach Zugriff auf Dinge bekommt, für die er, um es anders lösen zu können, mindestens Assemblerkenntnisse haben müßte. Denkbar wäre hier ein Diskettenmonitor, mit dessen Hilfe leicht Daten verändert werden können, ohne erst umständlich Datenkanäle öffnen zu müssen.

Auch Sprite-Editoren helfen Zeit sparen: Die Daten müssen nicht umständlich berechnet werden, sondern der Anwender erstellt die Sprites mit der Maus (fast wie in einem Zeichenprogramm) und das Editor-Programm wandelt anschließend die Bilder in nutzbare Grafikdaten um.

Alle diese Ideen haben viele findige Programmierer aufgegriffen und Programme geschaffen, die das Arbeiten mit dem Amiga erleichtern. Diese Programme werden allgemein als Tools und Utilities bezeichnet. Auf den folgenden Seiten präsentieren wir Ihnen eine Auswahl an verschiedenen Hilfsprogrammen (Tools und Utilities), die den Anwender bei seiner Arbeit mit dem Computer entlasten. Viele dieser Programme sind sicher auch für Sie interessant. Sie finden sowohl kommerzielle Programme, die in jedem gutsortierten Fach- und Versandhandel angeboten werden (die Bezugsadressen finden Sie beim Durchsehen der Hersteller- und Anbieteranzeigen), wie auch Public Domain-Programme. Public Domain, gerne auch als Freesoft bezeichnet, ist ein Weg, gute Software für wenig Geld zu bekommen, da die Freesoft-Disketten für einen geringen Betrag von 5 bis 10 Mark erhältlich sind. Sollten Sie Interesse an einigen dieser preiswerten Disketten bekommen, so können Sie die Adressen der Freesoft-Anbieter der Public Domain-Seite in dieser Ausgabe entnehmen.

Alles in allem hilft Ihnen diese Vorstellung sicher, daß Sie Programme entdecken, die Ihnen das Arbeiten erleichtern. Sie werden sich wundern, wie viel mehr Spaß das »computern« macht, wenn umständliche Handgriffe entfallen. Und damit können Sie sich mehr und intensiver Ihrem Hobby Computer widmen. (dm)

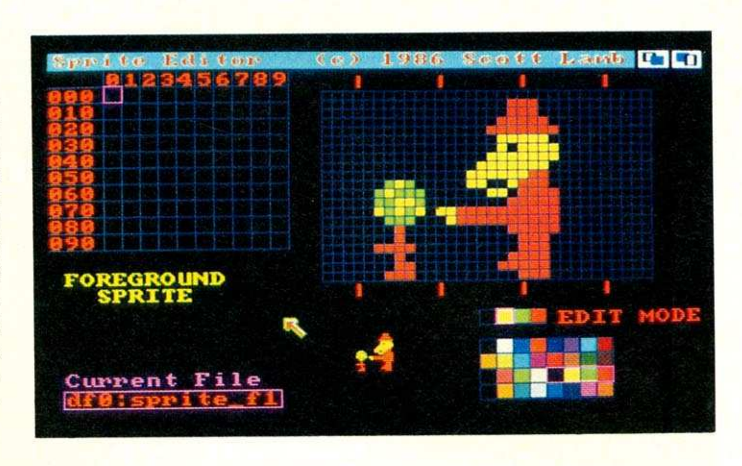


Systemverwaltung im Griff

Ein preiswertes Public Domain-Programm (Fish 70) ist der »AmigaMonitor«, der sicher für jeden Anwender interessant ist, der sich näher mit dem Amiga-System beschäftigt. Mit ihm hat man die Kontrolle darüber, welche Tasks (Prozesse) gerade mit welchen Prioritäten aktiv sind, wo sie im Speicher stehen und ob sie arbeiten oder warten. Auch die Verfügbarkeit und Größe des Fast-, Expansion- und Chip-RAM läßt sich ablesen. Weitere Stärken sind die Anzeige der Resources, der arbeitenden Devices mit zusätzlichen Angaben (etwa Versionsnummern), der aktiven Interrupts oder aller Libraries (System-Bibliotheken), die gerade geladen und geöffnet sind. Der Benutzer erfährt auch, welche Ports verwendet werden (etwa GamePort oder IDCMP) oder welche Files geöffnet sind. Eingebaut ist auch ein Speichermonitor, der den Speicherinhalt als Hex-Dump und im ASCII-Klartext ausgibt und so auch Einblicke in das Betriebssystem gewährt. Der Amiga-Monitor, der für den Systemprogrammierer eine Hilfe bei der Beherrschung des Systems darstellt, kann als ein nützliches Hilfsprogramm sicher in jede Tool-Sammlung aufgenommen werden.

Bunt gemischt

Speziell für Spieleprogrammierer dürfte der Freeware-Sprite-Editor »SpriteEd« von Nutzen sein. Das auf der Public Domain-Diskette Fish 40 zu findende Programm erlaubt das Editieren von zwei Sprites gleichzeitig. Jedes Sprite setzt sich aus vier verschiedenen Farben zusammen, von denen eine transparent ist, also der Hintergrundfarbe entspricht. In einem großen Rasterfeld können mit der Maus Punkte gesetzt und gelöscht werden. Das Sprite kann über ein Menü nach links oder rechts, oben oder unten verschoben sowie an der X- und Y-Achse gespiegelt werden. Im unteren Teil des Bildschirms ist gleichzeitig das gerade bearbeitete Sprite in Originalgröße zu sehen. Einzelne Sprites lassen sich auch in andere Felder kopieren, so daß durch weitere Manipulationen Bewegungssequenzen erstellt werden können.





Immer im Blickfeld

Wie praktisch kann es doch manchmal sein, wenn man immer die aktuelle Uhrzeit und das momentane Datum im Blickfeld hat. Wenn daneben noch eine Speicheranzeige steht, kann eigentlich nicht mehr erwartet werden. Es sei denn, man besitzt die Public Domain-»RSL-Clock« (AUGE 5). Dieses Programm ist mehr als nur eine Uhr mit Speicheranzeige. Von der RSL-Clock aus kann beispielsweise ein neuer CLI-Task gestartet werden. Auch läßt sich die Speicheranzeige in Chip- und Fast-RAM aufsplitten. Als brauchbarer Zusatz erweist sich ebenfalls die Möglichkeit, immer aktuell den Überblick darüber zu haben, wieviel Speicherplatz derzeit auf einem angeschlossenen Laufwerk (df0, df1, dh0, vd0) frei ist. Die RSL-Clock überdeckt im Normalfall einen Teil der Workbench-Menüleiste, so daß der Benutzer jederzeit die erwähnten Daten im Blickfeld hat.

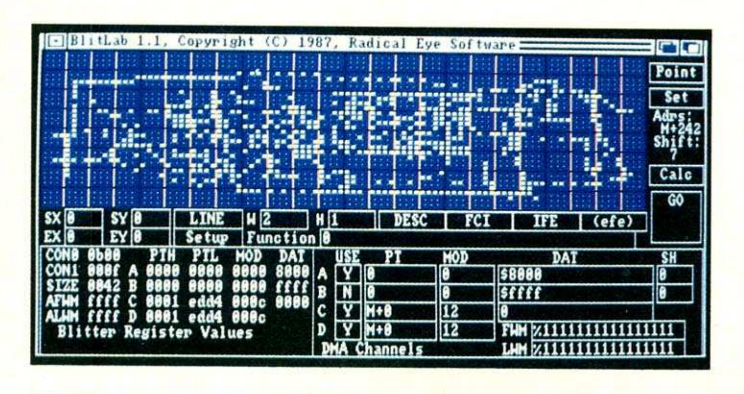


Tote Tasten werden lebendig

Der Amiga verfügt über zehn Funktionstasten, die im Normalfall nicht belegt sind. Wie schön wäre es doch, diesen Tastenblock mit Befehlen zu belegen, die häufig benötigt werden. Das Public Domain-Programm »FuncKeys« (Fish 106) erlaubt dies sogar auf fünf Ebenen. Das bedeutet, daß sich nicht nur die zehn Funktionstasten, sondern insgesamt 50 verschiedene Tastenbelegungen (in Verbindung mit SHIFT, ALT und den beiden Amiga-Tasten) nutzen lassen. Die Tastenbelegung arbeitet als eigenständiger Task im Hintergrund, so daß auch aus anderen Programm heraus die Befehle aktiviert werden dürfen. Dies geht sogar so weit, daß selbst ein A2000 mit PC-Karte diese Belegung vom PC-Modus aus ansprechen kann. Jede einzelne der 50 Funktionstasten läßt sich mit ganzen Befehlssequenzen belegen, so daß sich entweder Programme, CLI-Tätigkeiten oder ganze Befehlssequenzen auf Tastendruck aktivieren lassen. Das Programm eignet sich auch zum Einbinden in die Startdatei, da die Tastenbelegung automatisch beim Aufruf von FuncKeys mitgeladen und aktiviert wird. Dieses Hilfsprogramm sollte wegen seiner Vielseitigkeit und der damit verbundenen Arbeitserleichterung in die engere Wahl fallen.

Disketten durchleuchtet

Sind Sie neugierig, was sich alles auf Ihren Disketten befindet oder möchten Sie Dateien selbst verändern? Dann benötigen Sie einen Diskettenmonitor, um die einzelnen Bytes der Diskette zu manipulieren. Als Public Domain gibt es hierzu den Diskettenmonitor »Sectorama«, der sich auf der Fish-Disk 102 findet. Das Programm ist zwar nicht ganz so leistungsfähig wie kommerzielle Disktools, ist aber durch den geringen Preis von etwa 10 Mark, die man für die ganze Diskette bezahlen muß, lohnenswert. Außerdem ist Sectorama bis jetzt der einzige uns bekannte Diskmonitor, der auch Zugriffe auf (fast) beliebige Festplatten gestattet. Sie können mit diesem Utility einzelne Sektoren einlesen und sich deren Inhalte ansehen. Natürlich kann der Anwender diese auch manipulieren und wieder zurückschreiben.

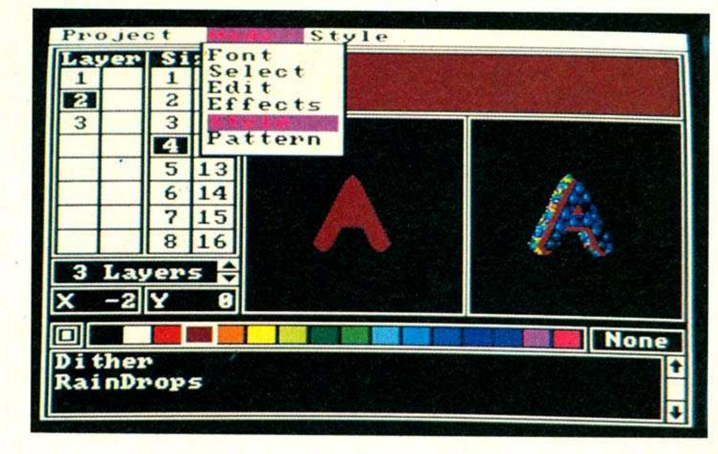


Blitterexperimente

Das auf Tornado 5 zu findende Freesoft-Programm »BlitLab« ist eigentlich kein Tool, sondern mehr ein Programm zum Experimentieren und Lernen. BlitLab dient dazu, die Funktion einer Bitplane kennenzulernen. Der Benutzer findet auf dem Bildschirm eine kleine Bitmap (96 x 32 Byte) mit den dazugehörigen Registern und kann diese nach Herzenslust manipulieren. So lassen sich Punkte setzen und löschen, wobei man verfolgen kann, wie sich die Werte in Abhängigkeit davon ändern. Es wird klar, wie einzelne Bits innerhalb der Plane errechnet werden und wozu ein Offset (Wert, der auf eine Adresse aufaddiert wird) gut ist.

Farbenvielfalt

Auf jeder Workbench-Diskette sind zwar schon einige alternative Zeichensätze enthalten (etwa Diamond oder Emerald). Für einige Anwendungen (Zeichen- oder Textprogramme) sind diese aber oft nicht ausreichend. Ein höchst leistungsfähiges Programm, mit dem sich Zeichen erstellen lassen, ist der inzwischen eingedeutschte »Calligrapher« (Test in AMIGA 10, Preis etwa 200 Mark). Der Editor kann Zeichen erzeugen, die maximal 256 x 160 Pixel groß sein dürfen und bis zu 16 Farben beinhalten. Diese Fonts lassen sich beispielsweise auch von Deluxe-Paint oder Prism verwenden, ebenso wie mit Deluxe-Paint erstellte und als IFF-File gespeicherte Zeichen übernommen werden können. Selbstverständlich kann der Calligrapher auch normale, einfarbige Fonts generieren, die auch in allen Programmen benutzt werden dürfen, die mit alternativen Fonts arbeiten können.



AMIGA Markt&Technik

PROGRAMM-SERVICE

Liga-Verwaltung für Sportprofis

LigaTab: Das Programm des Monats dient zur Verwaltung der verschiedensten Sportligen. Viele statistisch relevante Daten können komfortabel erfaßt und abgerufen werden. **Quatron:** Etwas ganz Besonderes für Spielefreaks ist dieses kurze C-Programm. Gute Grafik und hohe Spielmotivation sind seine Merkmale. **PlotIt:** Ausdrucke von Funktionen in fantastischer Qualität erhalten Sie mit diesem Basic-Programm. Die ideale Ergänzung zu dem Programm Kudiplo (Ausgabe 10/87). **MouseCreator und PointerMaker:** Ein Basic- und ein C-Programm zur Generierung von eigenen Mauszeigern. Damit kommt Leben in Ihre Maus.
Natürlich sind auch alle weiteren Programme aus der Rubrik Programmieren der Ausgabe 2/88 auf der Diskette enthalten.

Diskette für Amiga

Bestell-Nr. 48802 **DM 29,90*** (sFr 24,90*/öS 299,–*)
* Unverbindliche Preisempfehlung

Super-Spiel mit Sprache und toller Grafik

Domino: Ein Leckerbissen für alle Spiele-Fans ist dieses Programm des Monats. Sehr gute grafische Aufmachung und Sprachausgabe machen dieses Spiel besonders interessant. **Arraycomp:** Ein sehr nützliches Unterprogramm für alle Basic-Programmierer. Das Assemblerprogramm (in DATAs) findet gesuchte Strings bis zu 50mal schneller als ein Basic-Programm. **Julia:** Für alle Freunde schöner Grafik ist dieses Basic-Programm gedacht. Es stellt Juliamengen auf verschiedenste Weise auf dem Bildschirm dar. **Amiga-Sort:** Dieses Basic-Programm macht ein für allemal Schluß mit dem Chaos in Ihrer Diskettensammlung. Die Bedienung mit der Maus macht die Arbeit mit Amiga-Sort zum Vergnügen. **Disk.h:** Alle C-Programmierer brauchen dieses Programm, mit dem sogenannte »File Requester« durch einen Aufruf benutzt werden können. **Propgad:** Wer schon immer Schieberegler (Proportional Gadget) verwenden wollte, um Zahlen einzugeben, findet hier die nötigen Grundlagen .

Diskette für Amiga

Bestell-Nr. 48801 **DM 29,90*** (sFr 24,90*/öS 299,–*)
* Unverbindliche Preisempfehlung

Die Disketten aus dem Programm-Service erhalten Sie direkt bei Markt&Technik oder im ausgewählten Fachhandel



Weitere Angebote auf der Rückseite!

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0

Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656 · ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0.

---

ostscheckkonto Nr. des Absenders

mpfängerabschnitt

DM | Pf

r Postscheckkonto Nr.
4 199-803
eferanschrift und Absender
r Zahlkarte

Z Ort
erwendungszweck
**&T Buchverlag
rogramm-Service**

Meine Kunden-Nr.:

DM | Pf | für Postscheckkonto Nr. **14 199-803**

Absender der Zahlkarte

PSchA | Postscheckkonto Nr. des Absenders | Postscheckteilnehmer

Zahlkarte/Postüberweisung

Die **stark umrandeten Felder** sind nur auszufüllen, wenn ein **Postscheckkontoinhaber** das Formblatt als **Postüberweisung** verwendet (Erläuterung s. Rücks.)

DM | Pf | (DM-Betrag in Buchstaben wiederholen)

für Markt&Technik
Verlag Aktiengesellschaft
in **8013 Haar**

Postscheckkonto Nr.
14 199-803
Postscheckamt
München

Ausstellungsdatum | Unterschrift

Für Vermerke des Absenders

Postscheckkonto Nr. des Absenders

Einlieferungsschein/Lastschriftzettel

DM | Pf

für Postscheckkonto Nr. **14 199-803** Postscheckamt **München**

für Markt&Technik
Verlag Aktiengesellschaft
Hans-Pinsel-Str. 2
in **8013 Haar**

Postvermerk

PROGRAMM-SERVICE

Super-Kopierprogramm mit viel Komfort

DCopy: Unser Programm des Monats, ein Kopierprogramm, das alles bietet, was man sich nur wünschen kann. Einige Fähigkeiten: Bis zu vier Laufwerke werden verwendet, Mehrfachkopien, abschaltbares Verify und vieles mehr. **SpeedHc:** Eine sehr schnelle Hardcopyroutine für Schwarzweißausdrucke mit höchster Qualität. Leicht an andere Drucker anzupassen. **Sternenhimmel:** Ein unentbehrliches Werkzeug für alle Himmelsbeobachter. Das Programm zeigt alle Sterne und Planeten von jedem beliebigen Punkt der nördlichen Hemisphäre. **Checkie42:** Der Checksummer für alle Programmiersprachen von Assembler über Basic bis zu C. Ab dieser Ausgabe finden Sie bei jedem Listing die Prüfziffern. **Joy:** Ein sehr kurzes und schnelles C-Programm zur Abfrage des Joysticks. Es ist leicht in eigene Programme einzubinden. **Amiga-Shell:** Ein C-Programm, das Komfort ins CLI bringt. Editieren der Befehlszeile, Funktionstastenbelegung und Aliasnamen sind nur einige Fähigkeiten dieses fantastischen Programms.
Diskette für Amiga

Bestell-Nr. 48705 **DM 29,90*** sFr 24,90*/öS 299,–*

IFF-Grafik-Show für Anspruchsvolle

IFF-Diashow: Als Resultat des in Ausgabe 6-7/87 gestarteten Malwettbewerbs präsentieren wir Ihnen zwei randvoll bespielte Disketten, die mit den besten Bildern des Wettbewerbs (siehe Amiga-Magazin 10/87, Seite 129) bespielt sind. Die Bilder können entweder mit dem enthaltenen Diashow-Programm angesehen oder mit jedem gängigen IFF-Malprogramm geladen werden. Lassen Sie sich überraschen, welche hochwertigen und vor allem sehenswerten Grafiken unsere begabten Leser geschaffen haben.
Zwei Disketten für Amiga

Bestell-Nr. 49705 **DM 29,90*** sFr 24,90*/öS 299,–*

Berechnung realistischer Bilder mit 3-D-Effekt

Raytracing: Mit dem Programm des Monats können Sie fantastische Bilder berechnen lassen, wobei der Strahlengang des Lichts berücksichtigt wird. **Termin:** Sie vergessen nie wieder Termine mit diesem Programm, das die Daten der nächsten 15 Tage automatisch anzeigt. **Chain Reaction:** Ein spannendes Spiel, das viel Taktik erfordert und praktisch nur auf dem Computer realisierbar ist. Lösen Sie eine Kettenreaktion aus! **EHB-Demo:** Wie man 64 Farben im Lores-Modus benutzen kann, zeigt dieses C-Programm, das den Extra-Half-Brite-Modus verwendet. **Copper:** Auch von Basic aus kann der Copper für erstaunliche Effekte eingesetzt werden. Die Verwendung der nötigen Bibliotheken können Sie dabei erlernen. **Cursor Cycle:** Diese Utility macht Schluß mit dem unscheinbaren Cursor, indem es dessen Farbe ständig ändert. **MyCLI:** Eine unentbehrliche Hilfe für alle, die CLI-Fenster mit besonderer Voreinstellung benötigen. **MEd:** Das Schreiben von C-Programmen für Pull-down-Menüs verkürzt sich auf wenige Minuten. Mit einem weiteren Teil können Sie dann CLI-Befehle im Pull-down-Menü verwenden.
Diskette für Amiga

Bestell-Nr. 48704 **DM 29,90*** sFr 24,90*/öS 299,–*

Super-Malprogramme

Rainbow-Drawer: Dieses Programm des Monats bietet leistungsfähige Befehle und Funktionen, wie sie von professionellen Programmen bekannt sind: bis zu 32 Farben, alle Auflösungen, viele Befehle zum Zeichnen sowie FILL mit Mustern, BOW und anderem. **Turtle:** Mit dieser Befehlserweiterung verfügen Sie über die Grafikbefehle, die bei Logo bekannt und beliebt sind. **Fractals:** Dreidimensionale, realistische Gebirge mit Schattierung erzeugt dieses Programm. **Clouds:** Genauso wirklichkeitsnah wie die Gebirge, aber noch erstaunlicher, sind die Wolken, die Sie mit Clouds generieren. **Apfelmännchen:** Hiermit erzeugen Sie schöne Grafiken aus der beliebten Mandelbrot-Ebene. **Kudiplo:** Ein gutes, unverzichtbares Werkzeug für die Kurvendiskussion stellt »Kudiplo« dar. **Senso:** Testen Sie mit dieser Computer-Adaption des bekannten Spiels Ihr Gedächtnis! **Division:** Bis zu 32000 Nachkommastellen können durch dieses Programm berechnet werden. **Alert:** Alarme, zum Beispiel die bekannten Guru-Meditations, können Sie nun selbst programmieren. Das Programm ist in erster Linie für C-Programmierer aufschlußreich. **Border:** Lassen Sie den Fensterrahmen des CLI-Fensters einfach verschwinden! **SCD:** Mit diesem Utility können Sie den Pfadnamen in der Titelleiste des Fensters anzeigen.
Diskette für Amiga

Bestell-Nr. 48703 **DM 29,90*** sFr 24,90*/öS 299,–*

Disk-Operation vereinfacht

ExDir: Mit dem Programm des Monats der Ausgabe 8/9 sind alle Schwierigkeiten mit den Subdirectories endlich vorbei. **Hardcopy:** Diese nützliche Routine gestattet es, Teile des Bildschirms auszudrucken. **Bio:** Um Ihre körperliche und seelische Verfassung zu erfahren, können Sie dieses fantastische Biorhythmusprogramm einsetzen. **Mask:** Dieses Utility hilft Ihnen beim Herstellen komplexer Eingabemasken. **Freeman:** Den momentan verfügbaren Speicherplatz erfahren Sie mit dieser Routine im CLI-Window. **Vokabel:** Zum Lernen von Vokabeln ist dieses Programm ein sehr gutes Werkzeug mit viel Komfort. **SimpleSprite:** Wie man Sprites auf dem Amiga programmiert, sehen Sie in diesem Demoprogramm. **Vier gewinnt:** Wenn Sie gern spielen und strategisches Denken trainieren wollen, ist »Vier gewinnt« genau das Richtige für Sie. Weiterhin finden Sie auf der Diskette ein nützliches Tool zur Veränderung des CLI-Windows (InitCLI).
Diskette für Amiga

Bestell-Nr. 48702 **DM 29,90*** sFr 24,90*/öS 299,–*

Leckerbissen für C-Programmierer

Netzgrafik: Wie Sie auf dem Amiga dreidimensionale Netzgrafiken erstellen, zeigt dieses Programm in Amiga-Basic. **Disketten-Monitor:** Dieser Monitor, geschrieben in Amiga-Basic, bietet viele Möglichkeiten bei der Diskettenmanipulation. **Menü-Routinen:** Benutzen Sie die Betriebssystemroutinen von C zum Erzeugen von Pull-down-Menüs. **Polydraw:** Wie Sie die Grafik-Bibliothek in Amiga-Basic sinnvoll nutzen, zeigt dieses Programm. **Font-Designer:** Amiga-Basic-Programm, das Ihnen ermöglicht, eigene Zeichensätze herzustellen. **FTYPE:** In C geschriebene Druckroutine mit viel Komfort. **FCat:** Schnelle Directory-Routine. **Objekt-Editor:** Ein Sprite- und BOB-Editor mit komfortabler Maussteuerung.
Diskette für Amiga

Bestell-Nr. 48701 **DM 29,90*** sFr 24,90*/öS 299,–*

Endlich: Pixelgenaue Hardcopies

Drucker-Utility: Druckroutine zur pixelgenauen Ausgabe beliebiger IFF-Grafiken auf Epson-kompatiblen Druckern. **Terminalprogramm:** Amiga-Basic-Programm, das alle Grundfunktionen professioneller DFÜ-Software beinhaltet. **Schnelles Kopierprogramm:** Mit diesem C-Programm sind weniger Diskettenwechsel beim Kopieren mit nur einem Laufwerk notwendig. **CLI-Tool:** Die Bedienung des Amiga-CLI erleichtern drei Amiga-DOS-Dateien in der RAM-Disk. **Sprachausgabe:** Zahlen bis zu einer Größe von einer Milliarde spricht dieses Programm perfekt in deutsch aus. **Betriebssystem-Offset:** Alle Offsets der Amiga-Betriebssystem-Routinen für Besitzer des SEKA-Assemblers.
Diskette für Amiga

Bestell-Nr. 32704 **DM 29,90*** sFr 24,90*/öS 299,–*

* Unverbindliche Preisempfehlung

Sie suchen hilfreiche Utilities und professionelle Anwendungen für Ihren Computer? Sie wünschen sich gute Software zu vernünftigen Preisen? Hier finden Sie beides!
Unser stetig wachsendes Sortiment enthält interessante Listing-Software für alle gängigen Computertypen. Jede Woche erweitert sich unser aktuelles Angebot um eine weitere interessante Programmsammlung für jeweils einen Computertyp. Wenn Sie Fragen zu den Programmen in unserem Angebot haben, rufen Sie uns an: Telefon (089) 4613-640 oder (089) 4613-133.

Bei Fragen zu Bestellung und Versand der Programmservice-Disketten wählen Sie bitte Telefon (089) 4613-232. Bestellungen bitte an: Markt&Technik Verlag AG, Unternehmensbereich Buchverlag, Hans-Pinsel-Straße 2, D-8013 Haar, Telefon (089) 4613-0. Schweiz: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656. Österreich: Ueberreuter Media Handels- und Verlagsgesellschaft mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0. Microcomput-ique, E. Schiller, Fasangasse 24, A-1030 Wien, Telefon (0222) 785661;. Bücherzentrum Meidling, Schönbrunner Straße 261, A-1120 Wien, Telefon (0222) 833196. Bestellungen aus anderen Ländern bitte nur schriftlich an: Markt&Technik Verlag AG, Abt. Buchvertrieb, Hans-Pinsel-Straße 2, D-8013 Haar, und gegen Bezahlung einer Rechnung im voraus.

Bitte verwenden Sie für Ihre Bestellung und Überweisung die abgedruckte Postgiro-Zahlkarte, oder senden Sie uns einen Verrechnungs-Scheck mit Ihrer Bestellung. Sie erleichtern uns die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

Einlieferungsschein/Lastschriftzettel
(nicht zu Mitteilungen an den Empfänger benutzen)

Gebühr für die Zahlkarte
(wird bei der Einlieferung bar erhoben)
bis 10 DM — — 90 Pf
über 10 DM (unbeschränkt) 1,50 DM
Bei Verwendung als Postüberweisung gebührenfrei

Bedienen Sie sich der Vorteile eines eigenen Postgirokontos
Auskunft hierüber erteilt jedes Postamt

Feld für postdienstliche Zwecke

Hinweis für Postgirokontoinhaber:
Dieses Formblatt können Sie auch als Postüberweisung benutzen, wenn Sie die stark umrandeten Felder zusätzlich ausfüllen. Die Wiederholung des Betrages in Buchstaben ist dann nicht erforderlich. Ihren Absender (mit Postleitzahl) brauchen Sie nur auf dem linken Abschnitt anzugeben.
1. Abkürzung für den Namen Ihres Postgiroamts (PGiroA) siehe unten
2. Im Feld »Postgiroteilnehmer« genügt Ihre Namensangabe
3. Die Unterschrift muß mit der beim Postgiroamt hinterlegten Unterschriftsprobe übereinstimmen
4. Bei Einsendung an das Postgiroamt bitte den Lastschriftzettel nach hinten umschlagen

Abkürzungen für die Ortsnamen der PGiroA:
Bln W = Berlin West, Kln = Köln
Dtmd = Dortmund, Lshfn = Ludwigshafen am Rhein
Esn = Essen, Mchn = München
Ffm = Frankfurt am Main, Nbg = Nürnberg
Hmb = Hamburg, Sbr = Saarbrücken
Han = Hannover, Stgt = Stuttgart
Krlh = Karlsruhe

Für Mitteilungen an den Empfänger

Bestellung Programm-Service — Wichtig: Lieferanschrift (Rückseite) nicht vergessen!

Bestell-Nr.	Anzahl	x Einzelpreis	= Gesamtpreis

Summe bitte auf Vorderseite übertragen — Gesamtsumme:

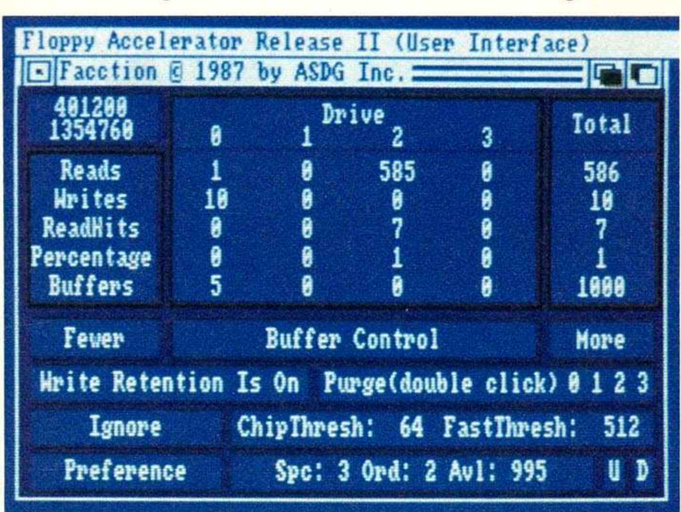
Für Grafiker unverzichtbar

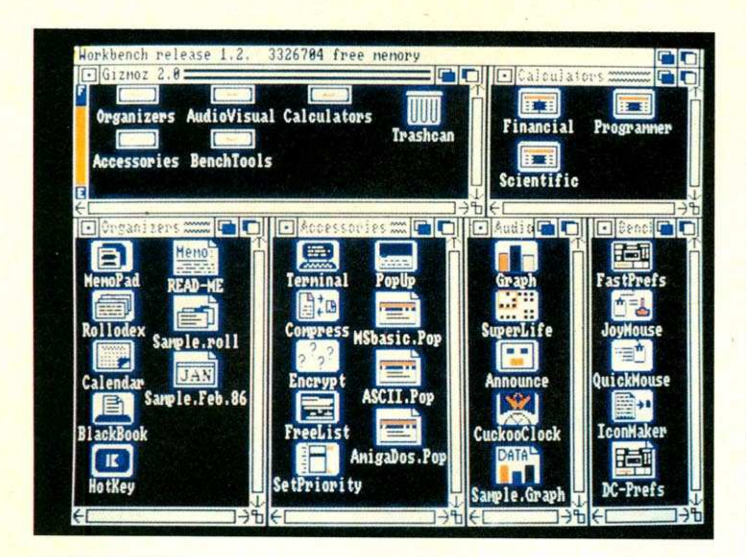
Wer sich viel mit Grafiken und Bildern auf dem Amiga beschäftigt und beispielsweise Titelbilder für Spiele oder ähnliches entwirft, wird an »Butcher« (Test in AMIGA 10 und dieser Ausgabe) nicht vorbeikommen. Dieses Programm ist im Grunde nur eine Sammlung von vielen grafikorientierten Hilfsroutinen, die zu einem einzigen Programm zusammengefaßt wurden. So existieren zum Beispiel Routinen, mit deren Hilfe sich die Zahl der Farben im Bild reduzieren läßt, ohne daß die Grafik hinterher sehr viel anders aussieht. Die freigewordenen Farben dürfen anschließend wieder neu belegt werden. Oder aber es tritt der Fall auf, daß für ein Spiel viele Grafiken gezeichnet werden müssen, aber aufgrund von (immer noch herrschender) Speicherknappheit kein Platz dafür vorhanden ist. Mittels Butcher läßt sich dafür die Zahl der Bitplanes verringern, das Bild nimmt bei fast gleichem Aussehen weniger Platz im Speicher weg.



Laufwerke werden schneller

Durch die umständliche, aber auch flexible Dateistruktur (mittels Pointern/Zeigern) von Amiga-DOS bedingt, benötigt der Amiga teilweise recht lange, bis beispielsweise ein Directory angezeigt wird. Dies liegt daran, daß die einzelnen Dateiblöcke durch Zeigerstrukturen verbunden sind und das Betriebssystem sich diese erst zusammensuchen muß. Eine Möglichkeit, dies zu beschleunigen, kann mit »FACC II« (Preis etwa 50 Mark, Test in dieser Ausgabe) genutzt werden. Das Programm benutzt einen Teil des Speichers (maximal den kompletten) dazu, bereits gelesene Strukturen und Verzeichniseinträge zwischenzuspeichern, so daß beim nächsten Zugriff auf die entsprechende Diskette die Struktur aus dem schnelleren RAM und nicht mehr von der langsamen Diskette gelesen wird. Um so mehr Speicher FACC II zur Verfügung gestellt wird, desto mehr Strukturen können im RAM gehalten werden und entsprechend schneller erfolgen Dateizugriffe. Sicher eine hilfreiche Erleichterung.



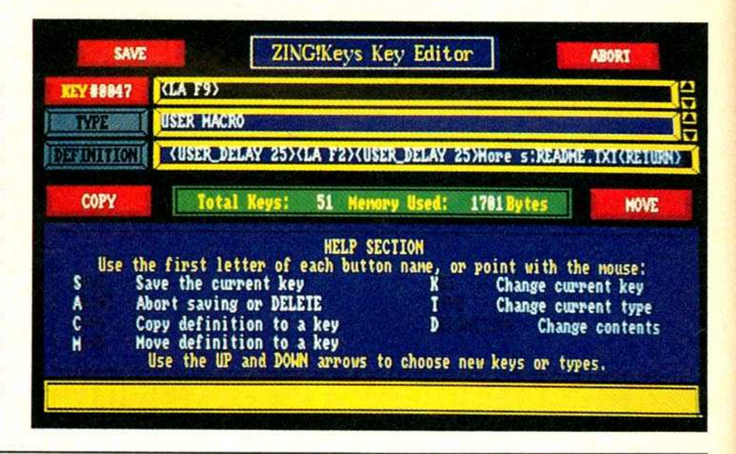


Vieles auf einen Streich

Die zwar schon betagte, aber trotzdem brauchbare Programmsammlung »Gizmoz« (Test in AMIGA 10, Preis etwa 150 Mark) beinhaltet eine ganze Reihe von Routinen, mit denen sich schon einiges anfangen läßt. Zum einen drei dem Workbench-Calculator ähnliche Taschenrechner, die mehrere Anwendungsgebiete (etwa zum Programmieren oder für wissenschaftliche Berechnungen) abdecken. Auch ein Editor für ASCII-Dateien oder ein Notepad-ähnliches Merkzettel-Programm fehlt nicht. Wer gerne Termine im Blick hat, freut sich sicher über den Kalender. Auch ein Programm zur Definition von Hotkey-Sequenzen (Makros) ist vorhanden. Dies bedeutet: Durch Drücken der Control- oder Alt-Taste in Verbindung mit einer beliebigen anderen Taste können vordefinierte Programmabläufe gestartet werden. Auch Programme lassen sich so aktivieren oder Batch-Dateien aufrufen. Durch die Vielzahl der verschiedenen Routinen ist diese Diskette sicher interessant, wenn es darum geht, sich eine Sammlung von Toolkits zuzulegen.

Noch mehr Funktionstasten

Sicher haben auch Sie sich schon einmal gedacht, wie praktisch es wäre, wenn verschiedene Tasten mit oft benötigten Befehlsfolgen oder Kommandos belegt werden könnten, so daß ein Druck auf eine Taste ausreicht, einen Vorgang auszulösen. »Zing!Keys« (Test in AMIGA 8/9) hilft hier: Mit diesem Programm kann jeder Taste, die in Verbindung mit einer der Sondertasten (etwa CTRL oder ALT) gedrückt wird, ein Befehl zugeordnet werden. So lassen sich beispielsweise durch Tastendruck Programme starten oder auch CLI-Tasks aktivieren. Auch die Möglichkeit, als »HotKeys« bezeichnete Makros (Befehlsfolgen) zu erzeugen, spricht für dieses Programm. Damit kann sich der Anwender ganze Kommandofolgen, die aus mehreren einzelnen Befehlen bestehen, generieren und dann auf Tastendruck aktivieren.

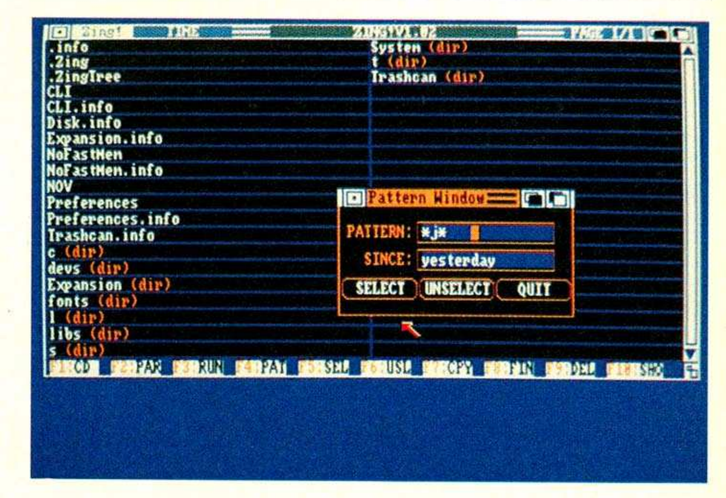


Das CLI im Griff

Um das Arbeiten mit der zweiten Amiga-Benutzeroberfläche CLI (Command Line Interface) zu erleichtern, greift man zum inzwischen mit deutschem Handbuch erhältlichen »CLI-Mate« (Test in AMIGA 8/9). Dateioperationen wie löschen, umbenennen oder kopieren können einfach mit der Maus aus einem übersichtlichen Menü ausgewählt werden. Dabei hat der Anwender das Directory des Quell- und Ziellaufwerks im Auge. CLI-Mate gestattet es, durch einen Druck auf die rechte Maustaste die Directory-Ebene zu wechseln und sich so leicht durch Verzeichnisebenen zu bewegen. Außerdem enthält das Programm eine Funktion, mit deren Hilfe IFF-Grafiken betrachtet werden können. Ein weiteres, komfortables Menü gestattet die Ausgabe beliebiger ASCII-Dateien auf Bildschirm oder Drucker.

Auf jeder Ebene

Ein weiteres Programm, das ebenso wie CLI-Mate Dateioperationen erleichtert, ist »Zing!« (Test in AMIGA 6/7). Es ermöglicht ebenfalls komfortabel Dateioperationen durch Mausbedienung oder Betätigen der Funktionstasten, wobei alle Verzeichnisse auf dem Bildschirm sichtbar sein können. Zing! stellt mehr Möglichkeiten als etwa CLI-Mate zur Verfügung. Von Zing! aus läßt sich jederzeit ein neues CLI-Fenster aktivieren oder ein Bildschirminhalt als IFF-Datei speichern. Die Funktionstasten lassen sich beliebig belegen und der aktuelle Screen kann über einen Drucker-Spooler auf einem angeschlossenen Drucker ausgegeben werden. Eine weitere, nützliche Funktion ist der Dimmer, der nach einer gewissen Zeit den Bildschirm »abschaltet«, so daß ein Einbrennen des Bildes in die Phosphorschicht des Monitors verhindert wird. Leider ist das Programm mit einem Preis von 189 Mark verhältnismäßig teuer.

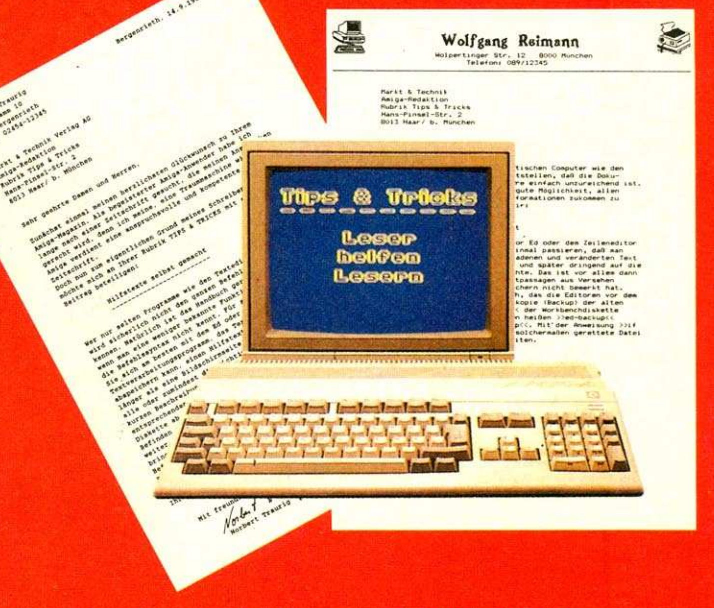


Der Bilderklau hilft, Grafiken zu bekommen



Sicher überkam auch Sie gelegentlich schon mal das Verlangen, beispielsweise eine besonders gut gelungene Spielegrafik in Ihre Bildersammlung aufzunehmen, um etwa ein besonders gut gelungenes Bild jederzeit ungestört zu betrachten oder weiterverarbeiten zu können. Doch wie bekommt man das Bild aus dem Spiel heraus? Bei den wenigsten Spiel- oder Anwendungsprogrammen sind die Grafiken in direkt verwertbarer IFF-Form gespeichert. Oft ist der Code des Bildes, also die Grafikdaten, bereits im eigentlichen Programm integriert. Und an diese Daten heranzukommen, ist wohl nur für geübte Freaks machbar. Glücklicherweise gibt es aber das speicherresidente »Grabbit« (Test in AMIGA 10, Preis etwa 70 Mark), das erlaubt, mit einer Tastenkombination die meisten Bilder aus den Programmen zu »entnehmen« und im IFF-Format zu speichern. Leider können auch mit Grabbit nicht alle Bilder »entfernt« werden, da es genügend Programme gibt, die gebootet werden müssen oder die speicherresidente Programme aus dem Speicher »werfen«. Eine andere Tastenkombination schickt den gerade sichtbaren Bildschirminhalt zum Drucker, wobei die bei Preferences eingestellten Druckertreiber benutzt werden. Auf dem nebenstehenden Bild können Sie sehen, wie sauber eine solche, mit Grabbit gemachte Hardcopy eines aktuellen Bildschirms aussehen kann. Ein weiteres, auf der Grabbit-Diskette mitgeliefertes Programm ist »AnyTime«. Aktiviert wird es ebenfalls mit einer Tastenkombination. AnyTime gestattet die jederzeitige Änderung der gerade auf dem Screen sichtbaren Bildschirmfarben. Dazu erscheint auf dem jeweiligen Screen (außer H.A.M. und EHB) ein Fenster mit entsprechenden Reglern zum Beeinflussen der Farben. Dies kann beispielsweise recht praktisch sein, wenn man einen Schwarzweiß-Monitor besitzt und schlecht lesbare Farben anpassen will. Das komplette Grabbit-Paket, das jedem grafikbegeisterten Amiga-Fan wärmstens empfohlen werden kann, ist ein nützliches Werkzeug für alle, die auch »fremde« Bilder in ihre IFF-Sammlung aufnehmen möchten.

Gesucht: Tips & Tricks zum Amiga



Halt, bevor Sie umblättern! — Haben Sie noch ein paar gute Tricks für den Amiga auf Lager? Dann sollte Sie unser Aufruf für die »Tips und Tricks« interessieren.

Schütteln Sie Ihre Asse aus dem Ärmel. Öffnen Sie Ihre Trickkiste für alle Leser. Senden Sie uns Ihre besten Ideen zum Amiga. Alle Griffe sind erlaubt.

- Hardware-Basteleien; genau richtig
- Software-Verbesserungen; fantastisch
- Anwendungs-Beispiele; super
- Spiele-Lösungen; oft der Retter in der Not
- Programmier-Kniffe; wunderbar
- Einsteigerhilfen; nicht wegzudenken
- Profi-Ratschläge; braucht jeder einmal

Es spielt keine Rolle, ob Sie selbst fortgeschrittener Programmierer oder ein Einsteiger sind. Sobald Sie etwas ausgeklügelt haben, schicken Sie es an uns. Wir geben Ihre Informationen weiter, damit Sie vielen Amiga-Fans helfen und neue Freunde gewinnen.

Für jeden Ihrer Beiträge, den wir veröffentlichen, erhalten Sie zusätzlich ein Honorar — damit Ihre Mühe auch belohnt wird. Also nichts wie ran an den Amiga. Experimentieren Sie, tüfteln Sie, suchen Sie nach den tollsten Tricks. Wenn Sie bereits ein paar gute Tips auf Lager haben — um so besser. Warten Sie nicht, bis ein anderer Leser auf dieselbe Idee kommt, denn jeden Beitrag können wir zunächst nur einmal drucken.

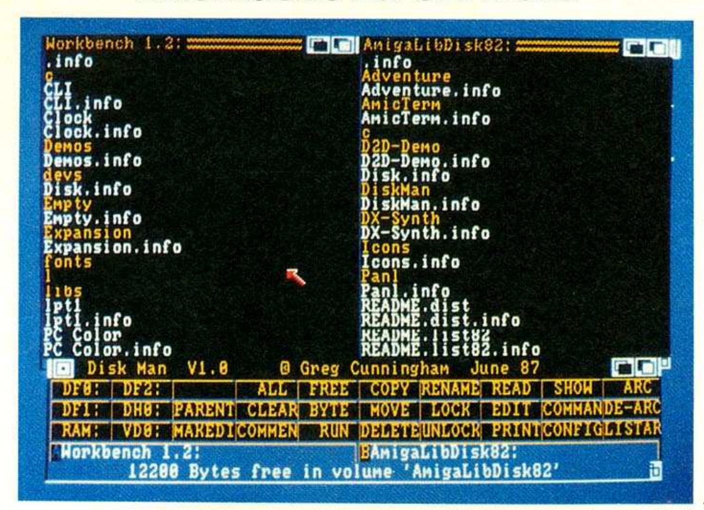
Schicken Sie Ihre »Tips und Tricks« an:

Markt & Technik Verlag AG
Redaktion AMIGA-Magazin
z.Hd. Ulrich Brieden
Aktion Tips & Tricks
Hans-Pinsel-Straße 2
8013 Haar bei München

Bei kurzen Texten reicht es, wenn Sie uns diese in einem einfachen Brief oder auf einer Postkarte zusenden. Bei Listings sollten Sie eine Programm-Diskette hinzufügen. Für Hardware-Basteleien sind Schaltpläne unbedingt erforderlich. Machen Sie also mit, erweitern Sie die Amiga-Fangemeinde durch Ihre Ideen.

(ub)

Alternative zu CLI-Mate



Ein Freesoft-Programm, das nützlich ist und Geld sparen hilft, ist »DiskMan« (Fish 82). Es bietet in etwa die gleichen Funktionen wie beispielsweise CLI-Mate oder Zing!, erleichtert also Dateimanipulationen, die sonst mit dem CLI erledigt werden müßten. Zum einen ist sowohl das Quell- wie auch das Ziellaufwerk in zwei getrennten Windows gleichzeitig auf dem Screen sichtbar. Gut daran ist, daß auch die Längen der einzelnen Dateien mit angezeigt werden und Subdirectories farblich abgehoben erscheinen. Zum anderen stellt DiskMan auch die Möglichkeit zur Verfügung, wie bei CLI-Mate IFF-Bilder anzusehen. Es ist also nicht mehr unbedingt erforderlich, daß man zum Betrachten der Grafiken ein Zeichenprogramm oder eine ähnliche Routine laden muß. Das Besondere ist aber die Funktion READ, mit deren Hilfe sich direkt Dateien auf der Diskette ansehen lassen. Das Programm folgt selbständig den einzelnen Dateiblöcken und zeigt auf dem Bildschirm sowohl die Hex-Werte wie auch den ASCII-Klartext an. Im Prinzip ist also ein kleiner Diskettenmonitor integriert, der aber nur einzelnen Datenblöcken folgen und deren Inhalte auf dem Bildschirm ausgeben kann. Als kleine Zugabe ist noch eine ARC- und eine DE-ARC-Routine implementiert. Mit dieser Funktion, die speziell für den DFÜ-Anwender interessant ist, können Dateien gepackt und wieder entpackt werden. Diese (standardisierte) Möglichkeit wird häufig dazu benutzt, gepackte Daten über das DFÜ-Netz zu senden, um die Übertragungszeiten klein zu halten. DiskMan kann des weiteren Kommentare an alle Dateien anfügen oder Files von einem Laufwerk auf ein anderes »moven«. Das bedeutet, daß die Datei auf das Ziellaufwerk kopiert und gleichzeitig im Quellverzeichnis gelöscht wird. Von DiskMan aus lassen sich auch CLI-Befehle aufrufen oder andere Tasks aktivieren, was sicher auch bei solchen Hilfsprogrammen nicht alltäglich ist, ebenfalls wie die Funktion »Lock« und »Unlock«, mit denen die Flags für Löschen und Überschreiben beeinflußt werden können. Alles in allem ist DiskMan eines der Programme, die in keiner Software-Sammlung fehlen sollten, da der Anwender damit preiswert seinen Computer besser im Griff hat.

Neue Dimensionen in Basic

Das Amiga-Basic besitzt viele mächtige Funktionen, aber es reizt doch sehr, alle Fähigkeiten des Amiga auszuschöpfen. Durch die Verwendung der vorhandenen Bibliotheken eröffnen sich neue Dimensionen. Wir zeigen, wie es funktioniert.

Der Amiga benutzt für seine Arbeit viele Bibliotheken (Libraries), die in C oder Assembler geschrieben wurden. In diesen Bibliotheken stehen jedoch keine Bücher, sondern Funktionen zu dem jeweiligen Bereich.

So finden Sie auf der Workbench-Diskette im Unterverzeichnis »libs« Dateien, die die Endung ».library« besitzen. Unter anderem stehen hier die Libraries für Icons (»icon.library«) und die verschiedenen Zeichensätze (»diskfont.library«). Die darin enthaltenen Funktionen können auch von Basic aus benutzt werden.

Leider gestaltet sich die Verwendung der Bibliotheken nicht ganz so einfach. Damit Basic die Routinen verwenden kann, muß es wissen, welche Parameter das Modul verlangt. Diese Angaben findet man in den Dateien auf der »Extras«-Diskette im Unterverzeichnis »fd1.2«, die die Endung ».fd« (»fd« heißt File Description, also Dateibeschreibung) besitzen. Aber dieses Format kann von Basic nicht benutzt werden. Deshalb muß der Benutzer diese Dateien noch konvertieren. Dazu verwendet man das Basic-Programm »ConvertFD« in Subdirectory »BasicDemos«.

Zuerst konvertieren...

Zu der Benutzung des Programms gibt es nicht viel zu sagen, aber ein kleines Problem gibt es doch. Nach dem Starten wird die Eingabe der zu lesenden ».fd«-Datei gefragt. Hier muß man den kompletten Pfad und die Endung ».fd« angeben, zum Beispiel:

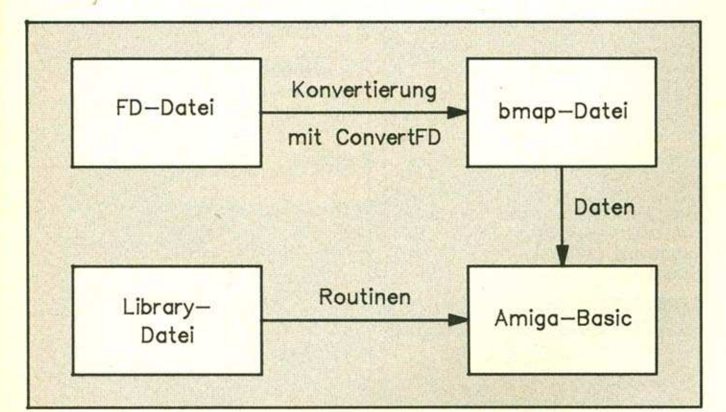
```
df1:fd1.2/graphics_lib.fd
```

Auch bei der Angabe der zu erzeugenden Datei muß ein kompletter Pfad und die Endung ».bmap« vorhanden sein.

```
df0:libs/graphics.bmap
```

Die Namen der Dateien müssen dieselben sein wie die der Library-Dateien, bis auf die Endung ».bmap«. Amiga-Basic sucht sie später im aktuellen Verzeichnis oder in der »libs«-Schublade.

Bei manchen Funktionen erfolgt die Ausgabe einer Warnung, da die Routine Parameter in einem Register verlangt, das Amiga-Basic nicht verwenden kann. Nach einiger Wartezeit sind dann die übersetzbaren Funktionen in der Datei enthalten. Aus diesem File kann Basic dann die benötigten Informationen entnehmen. Ein Problem ergibt sich noch, nämlich die Namensgleichheit von Basic-Befehlen und Funktionen, zum Beispiel bei »Read«. Das Programm »ConvertFD« stellt solchen Routinen ein »x« voran. Die Amiga-DOS-Funktion »Read« heißt dann also »xRead«.



Der Zusammenhang zwischen »File Descriptor«- und »Bit Map«-Dateien und den Bibliotheken

Wohlgemerkt stehen in den ».bmap«-Dateien nicht die Funktionen selbst, sondern nur ihr Offset (also die Stelle, an der sie zu finden sind) und die Parameter mit den entsprechenden Registern. Die Routinen selbst stehen in den Dateien mit der Endung ».library« wie oben beschrieben. Eine grafische Darstellung des Sachverhalts sehen Sie im Bild.

...dann benutzen

Nun sind wir endlich so weit, daß wir von Basic aus auf fast alle Library-Routinen zugreifen können. Zunächst muß das Basic aber noch erfahren, welche Funktionen einer Library einen Rückgabewert liefern. Dazu dient der Basic-Befehl DECLARE FUNCTION. Der Aufruf sieht prinzipiell so aus:

```
DECLARE FUNCTION Name (Parameterliste) LIBRARY
```

»Name« ist dabei ein gültiger Name einer numerischen Basic-Variable. Die verschiedenen Typen (%,&,! und #) können dabei auch angegeben werden. Es ergeben sich nämlich falsche Werte, falls zum Beispiel die Funktion einen doppeltgenauen Fließkommawert zurückgibt und Sie in Basic einen einfachgenauen Integerwert abholen. Dies kann zu großen Schwierigkeiten führen.

Die »Parameterliste« wird zwar von Amiga-Basic ignoriert, aber es ist guter Programmierstil, sie anzugeben.

Nur noch ein Schritt ist zu tun bis zur Benutzung der Funktion. Der Name der Bibliothek muß noch festgelegt werden. Dazu dient der Befehl »LIBRARY«. Passend zum obigen Beispiel, sieht das so aus:

```
LIBRARY "graphics.library"
```

Die Bibliothek wird geladen, sofern sie gefunden wird und der Speicher ausreicht. Andernfalls bekommen Sie die Fehlermeldung »File not found« oder »Out of memory«.

Sie können bis zu fünf Bibliotheken zur gleichen Zeit geöffnet haben, wodurch eine große Zahl von zusätzlichen »Befehlen« erreicht wird.

Jetzt endlich steht uns die Funktion zur Verfügung. Der Aufruf sieht so aus wie der eines Basic-Unterprogramms. Das Befehlswort »CALL« vor dem Funktionsnamen kann entfallen. Eine Ausnahme gibt es aber auch hier, steht der Aufruf hinter dem Befehlswort »ELSE« muß CALL erscheinen. Dies rührt daher, daß Amiga-Basic sonst ein Label erwartet.

Wollen Sie die Funktion benutzen, schreiben Sie:

```
CBump (Parameter)
```

Es existieren aber auch Routinen, die einen Wert zurückgeben. Um diesen Wert zu erhalten sieht der Aufruf dann so aus:

```
erfolg%=Execute&(SADD("dir >ram:temp"+CHR$(0)),0,0)
```

Bei diesem Beispiel wird der Befehl »dir« von der Diskette ausgeführt. »erfolg%« enthält nach der Durchführung einen Wert, der angibt, ob alles funktioniert hat. Übergeben wird die Adresse der Zeichenkette »dir df0:«. Vorher wird noch der ASCII-Wert 0 angehängt. Dies muß geschehen, da in der Programmiersprache C immer nur Zeiger auf Strings übergeben werden und diese mit dem ASCII-Wert 0 enden müssen. Die Funktion »Execute« muß allerdings vorher mit

```
DECLARE FUNCTION EXECUTE& LIBRARY
LIBRARY "dos.library"
```

vorbereitet werden. Außerdem muß man das Amiga-Basic von einem CLI-Fenster aus gestartet haben.

Am Ende Ihres Programms sollten Sie auf keinen Fall vergessen, die geöffneten Bibliotheken wieder zu schließen. Zum Schließen verwenden Sie den Befehl

```
LIBRARY CLOSE
```

Nun wissen Sie wie man von Amiga-Basic die Bibliotheken benutzen kann und wissen auch, daß dies nicht ohne Probleme ist. Bevor Sie also anfangen, müssen Sie sich die einzelnen Funktionen ansehen. Beherrschen Sie dann die benötigten Funktionen, stehen Ihnen neue Dimensionen in der Basic-Programmierung offen. (rb)

Fortsetzung von Seite 125

Emulieren...

1541-Diskette wie jede »normale« C 64-Diskette direkt bearbeitet werden.

Wird nun ein Amiga-Laufwerk statt der 1541 verwendet, kann natürlich auch das Anschlußkabel für das Original-C 64-Laufwerk ausgesteckt und das Druckerkabel wieder eingesteckt werden, was aber nur bei ausgeschaltetem Amiga geschehen sollte. Jetzt wird der Amiga-Drucker vom 64-Emulator als Drucker für die C 64-Programme verwendet. Das ist ein entscheidender Vorteil des 64-Emulator. Der GO 64! erlaubt es lediglich, einen Drucker an die 1541-Floppy anzuschließen. Somit bräuchte der GO 64!-Besitzer eigentlich zwei Drucker, um seinen Amiga voll ausnutzen zu können.

Die 1541 wird simuliert

Prinzipiell ist es sehr praktisch, eine Amiga-Floppy als Ersatz für die 1541 verwenden zu können — allein schon wegen der 3506 freien Blöcke, die dem »C 64«-Benutzer auf einer ansonsten leeren Diskette zur Verfügung stehen. Die Unterverzeichnisse des Amiga-Betriebssystems können allerdings mit dem 64-Emulator nur mit einigen Tricks verwendet werden. Man könnte diesem Problem zwar mit einigen »Assign«-Befehlen in der »startup-sequence« der 64-Emulator-Diskette abhelfen, aber wer verändert schon gerne seine Originaldiskette?

Bei der Arbeit mit den Amiga-Disketten tritt nur ein Problem auf: Da das Amiga-Laufwerk keine bei Fehlern blinkende Lampe besitzt, merkt der Benutzer manchmal nicht, wenn die Floppy einen Fehler macht. Besonders unpraktisch ist, daß der Emulator auch beim Speichern auf schreibgeschützte Amiga-Disketten keine Fehlermeldung ausgibt. Tritt auf der Amiga-Diskette ein Lesefehler auf, so kommt es zum Absturz des Emulators, da die erscheinende Dialogbox des Betriebssystems mangels Mauszeiger nicht mehr abgeschaltet werden kann.

Sehr interessant für DFÜ-Freunde dürfte der 64-Emulator dadurch werden, daß er alle am seriellen Port des Amiga angeschlossenen Geräte unterstützt. Ein Terminalprogramm kann also im C 64-Modus ganz normal mit dem Modem am Amiga arbeiten. Erstaunlich ist, daß dabei Übertragungsraten bis zu 19200 Bit pro Sekunde möglich sind — der Original-C 64 schafft nicht einmal 4800 Bit pro Sekunde.

Bei beiden Emulatoren kann es besonders beim Ausprobieren neuer Programme dazu kommen, daß der simulierte C 64 »abstürzt«. Deshalb wurde auch bei beiden die Tastenkombination <RUN/STOP RESTORE> des C 64 auf zwei Tasten des Amiga umgesetzt. Leider ist diese Tastenkombination bei vielen C 64-Programmen unwirksam. Dem Besitzer der getesteten GO 64!-Version bleibt in diesem Fall nichts anderes übrig, als einen Reset des Amiga auszulösen und den Emulator wieder neu zu starten. Laut Hersteller wird das aber in einer der nächsten Programmversionen geändert. 64-Emulator-Besitzer haben es da besser — mit <CTRL ALT HELP> können sie in den meisten Fällen den simulierten C 64 zurücksetzen.

Der 64-Emulator wird mit einem 16seitigen, der GO 64! mit einem 19seitigen englischen Anleitungsheft geliefert. Die beiden Anleitungen sind verständlich geschrieben; beim GO 64! hätten aber einige Details mehr sicher nicht geschadet. Das 64-Emulator-Anleitungsheft ist teilweise etwas unübersichtlich; ihm fehlt eine Grafik der Tastaturbelegung.

Obwohl beide Emulatoren ihre Vorteile haben, ist man insgesamt mit dem 64-Emulator sicher wesentlich besser beraten. Allein schon die Nutzungsmöglichkeit von Amiga-Floppy und -Drucker macht dieses Programm dem GO 64! überlegen. Doch auch kleinere Details wie Tastaturabfrage und Kontrollfenster sind beim 64-Emulator deutlich besser gelöst, obwohl auch hier noch einiges verbessert werden könnte. Die höhere Geschwindigkeit und Kompatibilität tun ein übriges. Ein vollwertiger Ersatz für einen Commodore 64-Computer kann allerdings keiner der beiden Emulatoren sein. (Andreas Lietz/jk)

AMIGA-WERTUNG

Software: GO 64!

4,9 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	X	X				
Dokumentation	X	X	X	X		
Bedienung	X	X				
Erlernbarkeit	X	X	X	X	X	
Leistung	X	X				

Fazit: Der GO 64! fällt hinter seinem Konkurrenten stark zurück, da er wesentlich langsamer und in vielen Teilbereichen unpraktischer als der 64-Emulator ist.

Positiv: Voreinstellung der Floppy-Geräteadresse auf 8; Belegung von <SHIFT RUN/STOP> zum Laden des ersten Programmes der Diskette.

Negativ: Kopierschutz; Amiga-Floppy und -Drucker werden nicht unterstützt; langsame Tastaturabfrage und Scrolling; Maus muß ausgesteckt werden; kein Reset des Emulators; geringe Geschwindigkeit; fehlende Commodore-ROMs; zur Parametereinstellung muß der Emulator verlassen werden.

DATEN

Produkt: GO 64!
Preis: etwa 200 Mark
Hersteller: Software Insight Sys.
Anbieter: gutsortierter Fach- und Versandhandel

AMIGA-WERTUNG

Software: 64-Emulator

8,1 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	X	X	X	X	X	
Dokumentation	X	X	X	X		
Bedienung	X	X	X	X		
Erlernbarkeit	X	X	X	X	X	
Leistung	X	X	X	X		

Fazit: Der 64-Emulator ist in der Lage, die meisten C 64-Programme mit einer akzeptablen Geschwindigkeit zu verarbeiten. Software-Probleme ergeben sich vor allem im Bereich schneller Sound- und Grafikprogrammierung und bei den meisten Kopierschutzmethoden auf dem C 64. Vor allem die Verwendung der Amiga-Peripherie ist hervorzuheben.

Positiv: Amiga-Peripheriegeräte werden unterstützt; Simulation einer 1541; Kontrollfenster; Simulation der C 64-Mäuse und RAM-Erweiterung.

Negativ: Kopierschutz; Kontrollfenster nur mit Maus zu nutzen; Sprites flimmern; langsame Übertragung auf Amiga-Format; Fehler beim Arbeiten mit Amiga-Disketten nicht offensichtlich; nur eine 1541-Emulation pro Diskette; Arbeit mit Unterverzeichnissen unpraktisch; Absturz bei Lesefehlern auf Amiga-Disketten.

DATEN

Produkt: 64-Emulator
Preis: 148 Mark
Hersteller: Readysoft Inc.
Anbieter: Compustore, Fritz-Reuter-Str. 6, 6000 Frankfurt, Tel. 069/567399

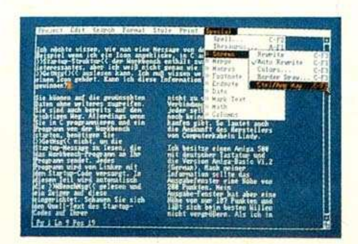
Inserenten

Preiswerte Alternativen



Public Domain, oft auch als Freeware oder kurz PD bezeichnet, ist ebenso wie Shareware eine preiswerte Alternative zu teuren, kommerziellen Programmen. Auf inzwischen über 300 verschiedenen Disketten sind jede Menge Routinen enthalten, die durchaus mit professionellen Programmen mithalten können. Deshalb finden Sie in der nächsten Ausgabe eine Vorstellung der besten PD-Programme, was sie leisten und kosten, ob sie ein vernünftiger Ersatz für kommerzielle Produkte sind, Tips und Tricks zum Umgang mit Public Domain und Hintergrundinformationen zu Anbietern und Freesoft-Serien.

Moderner Vierkampf



Sieht man einmal von Unterhaltungsprogrammen ab, so ist die Textverarbeitung eine der häufigsten Anwendungen auf dem Amiga. Auf dem deutschen Markt werben zur Zeit vier große Textverarbeitungen um die Gunst der Amiga-Anwender. Das schon länger auf dem Markt befindliche Prowrite hat Konkurrenz durch Vizawrite Desktop bekommen. Beide Programme können mehrfarbige Grafiken in einen mit den Zeichensätzen des Amiga geschriebenen Text einbinden. Sie wenden sich also an Anwender, denen es besonders auf die optische Gestaltung des Textes ankommt. Bei Becker-Text und Word-Perfect handelt es sich um Textverarbeitungen im klassischen Sinne. Ohne großartige Grafikfunktionen präsentiert sich einerseits Becker-Text von Data Becker mit niedrigem Preis und vom Amiga gewohntem Bedienungskomfort, andererseits Word-Perfect mit dem am PC üblichen Leistungsumfang. Wie betriebssicher arbeiten diese Programme? Gibt es das beste Textverarbeitungsprogramm für den Amiga? In unserem Vergleichstest finden Sie Antworten auf diese Fragen.

Lattice kontert

Der ständige Kampf um den schnellsten und besten C-Compiler auf dem Amiga ist in vollem Gange. Lattice hat seine neueste Version 4.0 ins Rennen geworfen. Wir haben den Compiler unter die Lupe genommen und festgestellt was er zu bieten hat. Sie dürfen schon jetzt gespannt sein, mit welchen Daten der Compiler aufwartet.

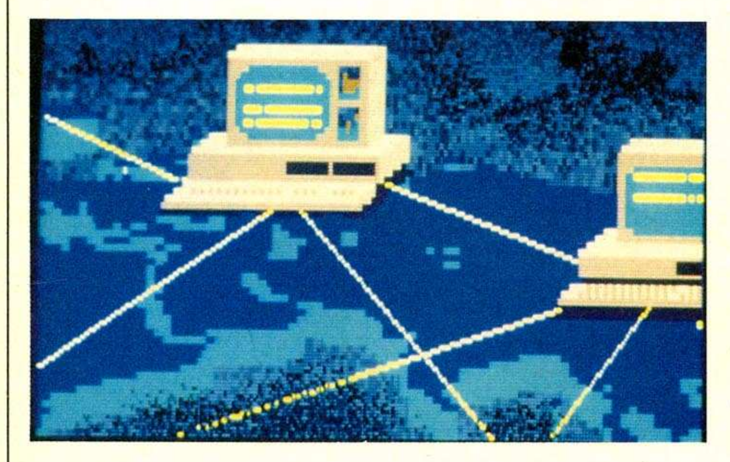
Aber auch viele andere neue Programme müssen zeigen wie gut sie sind. Für Musikfreunde ist Synthia von Interesse, da es bis jetzt unerreichte Fähigkeiten auf dem Amiga bietet. Eher für Techniker und Elektroniker ist Newio gedacht, ein Platinenlayoutprogramm. Für welchen Einsatzbereich es geeignet ist, erfahren Sie im Test.

Vorschau 3/88

AUSSERDEM IN DER NÄCHSTEN AUSGABE
- PROGRAMM DES MONATS: BOOTGIRL FÜR EIGENWILLIGE VORSPÄNNE
- START EINES NEUEN GRAFIKWETTBEWERBS
- IM HARDWARETEST: WIE GUT IST DER MARVIN-SCANNER
- WAS BRINGT DER DIGITIZER DIGIPIC?
- VORSCHAU AUF DIE CeBIT
- VIELE LISTINGS ZUM ABTIPPEN UND LERNEN
- MITMACH-KURSE: C UND GRAFIK

Die nächste Ausgabe erscheint am 24.2.1988. Erhältlich bei Ihrem Zeitschriften-Händler.

DFÜ: Computer am Draht



Datenfernübertragung öffnet Ihnen eine Informationsquelle rund um den Globus. Mit einem Telefonanschluß, einem Akustikkoppler oder einem Modem sowie der richtigen Software können Sie mit Ihrem Amiga weltweit mit anderen Computern und deren Besitzern in Verbindung treten. Tauschen Sie Programme und Daten aus, holen Sie Informationen aus Datenbanken und Mailboxen, oder spielen Sie Computerspiele mit einem Partner irgendwo in der Ferne.

In der nächsten Ausgabe erfahren Sie alle Einzelheiten zum Thema DFÜ. Wir geben in einem Artikel für Einsteiger alle wichtigen Informationen, um DFÜ auf dem Amiga zu betreiben. Darin werden Emulationen und auch die verschiedenen Übertragungsprotokolle erklärt. Der Test zweier Koppler, die mit 1200 Bit/s arbeiten, wird auch Profis interessieren.

68000er-Systeme perfekt programmieren in Assembler und C

Für Ihre Anforderung verwenden Sie bitte nebenstehende Bestellkarte.

Erfolgreiches Programmieren von 68000er Systemen in Assembler und C

durch

- detaillierte Baustein- und Betriebssystembeschreibungen
- Trainingsprogramme für perfekte Programmierung
- programmierte Unterweisung in Assembler und C
- mit Bibliotheken und Programmsammlung



Ab sofort hilft Ihnen dieses neue Nachschlagewerk,

- effiziente Problemlösungen auf **Betriebssystemebene** zu realisieren
- anspruchsvolle **Anwenderprogramme** zu entwickeln
- mit ausgefeilter **Bausteinprogrammierung** neue Anwendungsbereiche zu erschließen.

Die wichtigsten Themen auf einen Blick:

- **detaillierte Hardware-Beschreibungen** der Prozessorfamilie 68000 und ihrer 8-, 16-, 32-Bit-Peripheriebausteine mit technischen Kennwerten, Pinbelegung, internen Architekturen, Befehlsbeschreibungen; Zudem erhalten Sie Hinweise für optimale Kombinationen wie zwischen MMC 68461/CPU 68012 und Weiterentwicklungen;
- **Betriebssysteme:** Sie lernen die Strukturen der 68000er-Betriebssysteme (wie GEM-TOS, OS-9 und Unix) kennen und erfahren, wie Sie die Systemroutinen zur Optimierung Ihrer Assembler- und C-Programme nutzen;
- **Softwareengineering:** die optimale Vorgehensweise von der Problemanalyse über die Codierung bis hin zur abschließenden Dokumentation;
- **Programmierkurse für effiziente Anwender-, System- und Bausteinprogrammierung in Assembler und C.** Damit verfügen Sie zugleich über sofort einsetzbare Routinen, z. B. zur Drucker- und Bildschirmansteuerung oder GEM-TOS-Utilities;
- **die C-Compiler-Werkstatt:** Schritt für Schritt programmieren Sie selbst einen C-Compiler inkl. Funktionsbibliotheken (Ein-/Ausgabe, Statistik/Mathematik...);
- **Assembler-Makrobibliotheken** (Stringverarbeitung...), **Anwender- und Funktionsbibliotheken in C** (Dateiverwaltung u. a.);
- **Tips und Utilities** wie Schnittstellen- und Backup-Routinen, Fensterverwaltung;
- **bewährte Musterlösungen** wie zu Rechnerkopplung und Multitasking;
- **Anleitung für raffiniertes Systemtuning,** z. B. durch Einsatz schnellerer Peripheriebausteine/Höhertakten der CPU;
- **Praxiswissen zu speziellen Einsatzbereichen** wie Messen, Steuern, Regeln. Mit den ausführlichen Anleitungen erstellen Sie z. B. ein universelles MSR-System mit eigener Peripherie oder programmieren EPROMS;

Dieses Werk veraltet nie
Unsere 68000er-Fachredaktion versorgt Sie regelmäßig und zuverlässig mit aktuellen Informationen über Hardware- und Betriebssystem-Weiterentwicklungen, weiteren Utilities und Musterlösungen sowie Bibliothekserweiterungen in Assembler und C.

Fordern Sie noch heute an:

Erfolgreiches Programmieren von 68000er-Systemen in Assembler und C

stabiler Ringbuchordner, Format DIN A 4, Grundwerk ca. 400 Seiten. Bestell-Nr. 3400, Preis: DM 92,– (erscheint ca. 1. Quartal 1988)

Alle 2–3 Monate erhalten Sie Ergänzungsausgaben zum Grundwerk mit je ca. 120 Seiten zum Seitenpreis von 38 Pfennig (Abbestellung jederzeit möglich).

